FLOW RATE MODULATED PERIODIC BACKFLUSH TO IMPROVE DEAD-END FILTRATION

A Dissertation
Presented to
The Academic Faculty

by

Aaron Christopher Enten

In Partial Fulfillment
of the Requirements for the Degree
BioEngineering in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
May 2019

FLOW RATE MODULATED PERIODIC BACKFLUSH TO IMPROVE DEAD-END FILTRATION

Approved by:

Dr. Todd A. Sulchek, Advisor
School of Mechanical Engineering
Georgia Institute of Technology

Dr. Manu Platt
School of Biomedical Engineering
Georgia Institute of Technology
Emory University

Dr. Craig R. Forest
School of Mechanical Engineering
Georgia Institute of Technology

Dr. Fatih Sarioglu
School of Electrical and Computer Eng.
Georgia Institute of Technology

Dr. Wilbur A. Lam
School of Biomedical Engineering
Georgia Institute of Technology
Emory University

Date Approved: March 26, 2019

I dedicate my thesis work to my family, friends, peers, undergrads, volunteers, and my advisor, Todd Sulchek; without their influence, I would not have developed into the researcher I am today. The support I have received across the board has challenged me to consistently improve myself, my work, and especially my management skills.

Thank you to my mother, father, brother, and sister for their support and encouraging me to pursue all my goals, academic or otherwise.

Thank you to my advisor, Todd Sulchek, for allowing me to pursue a novel thesis topic outside of the core research of the Sulchek BioMEMS laboratory and to enable my pursuits outside of everyday laboratory work

Special thanks to Laura Paige for always having my back and without whom I would not have survived the program or known what to do and where to go

Special thanks to all the undergraduate and high school volunteers who helped collect data and run experiments

ACKNOWLEDGEMENTS

I acknowledge the following for their contributions:

- Todd Sulchek, for his guidance, advice, and support throughout my graduate career.
- My thesis committee, Craig Forest, Wilbur Lam, Manu Platt, Fatih Sarioglu, for their strategic insight, critiques, and motivation to pursue this project in depth.
- The research volunteers who have worked with me since I joined the Sulchek BioMEMS lab for their hard work and scientific contributions: Matthew Leipner, Zach Matthews, Matthew Lemons, Jared Li, Antonio (Jun) Lee, Sami Belhareth, Emma Soetebier, Brandon Holt, Brandon Lo, Brighton Ancellin, Sebastian Khaf, Yusen Zhu, Shihong Cheng, Seko Nagae, Yuzhe Peng, Do Hyun Ahn, Gookhee Lee, Joohn Su Han, Sung Woo Cho, Caleb Baldwin, Andrew Hong, Lillian King, Vikram Varadarajan, Elizabeth Katafias, Sonia Cillero, Nehemiah Elias, Nicole Diaz.
- The Undergraduate Research Leaders who demonstrated commitment beyond the call to action: Antonio Lee, Matthew Leipner Lillian King, Elizabeth Katafias
- The Sulchek Lab, especially Michael Bellavia, Nicholas Stone, Katily Ramirez, Sebastian Kahf, Shawn Newlan, Tom Bongiorno, and Elizabeth Campbell for their experimental contributions, advice, and guidance, and Alan Liu, Anna Liu, Katie Young, Ahmad Haider, Dan Potter, Kathryn Murray, Bushra Tasadduq,
- Ryan Ackman for his assistance with tibial sample acquisition and micro-CT imaging.
- Michael Griffin for his assistance with whole blood acquisition and treatment.
- Eric Snider for his training and guidance using the Attune automated flow cytometer.
- Joscelyn Mejias for identifying and providing the a549 lung epithelial cultures.

- Fabrice Bernard for his assistance with experimental design and data collection for decellularized muscle ECM and collagen sponge seeding.
- Joshua Hooks for his assistance with training and preparing students for APTES bonding polycarbonate filters into microfluidic devices.
- Cameron Yamanishi for his help in particle position and movement analysis
- Core facilities staff Nadia Boguslavsky, Sommer Durham, Allen Echols, and Andrew Shaw for assistance with flow cytometry, inventory storage, and microscopy.
- Georgia Institute of Technology's Technology Transfer Office for prior art identification, claims writing, and patent filing.
- Technological Innovations: Generating Economic Results (TI:GER) program at Georgia Tech and Emory University for funding portions of the research and for discovering market potentials of the research.
- Jermaine Fanfair, Alexander Cohen, Rudy Banerjee, and Ariel Winawer for motivating real-world applications, for organizing customer interviews, and for identifying potential markets for the research.
- National Institute of Health Cell and Tissue Engineering T32 Training Grant for funding portions of the research and aiding the investigation into applications of the research.
- Elayne Ashley for offering her unending support and encouragement, especially during these past months while completing my thesis.
- Diane, Jay, Garrett, and Lauren Enten for everything they have done and will do to support me on my journey through life and for instilling in me a drive to pursue my passions and education with the fervor required to take on the most difficult tasks.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

Symbol	Definition	Units
A_m	effective membrane area	m^2
C_b	bulk solute concentration	kg/m^3
$C_{b,0}$	initial bulk solute concentration	kg/m^3
∂z	membrane thickness	m
f	fraction of solute present as aggregates	kg/kg
f'	fraction of solute which contribute to growth of blockage	kg/kg
$f'R'$	rate of increase of solute layer resistance with time	$m^{-1}s^{-1}$
J	solute flux across membrane	kg/m^2s
k	membrane permeability	kg/m
n	cycle number	-
ΔP,TMP	transmembrane pressure	Pa
Q_0	initial volumetric flow rate	m^3/s
R_m	resistance of clean membrane	m^{-1}
$R_{P,0}$	initial resistance due to solute deposit membrane	m^{-1}
t	time	s
t_p	time at which membrane region is first blocked by solute	s
V	instantaneous volume processed	m^3
V_R	backflush volume	m^3
V_T	total volume to be processed	m^3
α	pore blockage parameter, membrane area blocked/unit mass bead brought to membrane surface	m^2/kg
μ	fluid viscosity	Pa-s

SUMMARY

Dead-end filtration using patterned microsieves, fiber meshwork, and membranes of various materials is a standard technique to isolate desired particles of various sizes and is used in clinical and laboratory settings for both therapeutic and diagnostic applications. Both biological and physical suspensions can be filtered to yield high purity and enrichment at a high throughput. Unfortunately, dead-end filters are especially susceptible to fouling, which leads to lower recovery percentage and yield as a direct result. High recovery percentages, enrichment, and throughput determine the success of sorting technologies. A process which could stop fouling, reintegrate the fouling material into the bulk flow, and allow for further processing could be used to improve the recovery percentage of dead-end systems increasing their success and use.

The purpose of this research is to determine if novel flow profiles with variable duty cycles can reduce fouling and improve permeate flux without substantial tradeoffs to processing time by controlling forward volume flow rate, reverse volume flow rate, and the time spent in each phase. This objective was accomplished through 1) the development of pulse modulated (PM) periodic backflush using a square wave duty cycle fluid flow control systems to interrupt membrane fouling, 2) the development of a model to better understand how experimental results compare to what might be achievable, 3) the optimization of amplitude and frequency to reduce and minimize costs to throughput by increasing yield, and 4) demonstration that the control algorithms can be applied to important applications of particle purification, cell enrichment, and scaffold seeding. These practical tests also help to define the conditions upon which the developed methods

optimally apply. Ultimately, this thesis work established that pulse modulation is an effective technique to interrupt fouling and reintegrate the cake into the bulk flow to improve the recovery percentage of both microparticle and cellular products while minimizing the anticipated costs to throughput.

CHAPTER 1. INTRODUCTION

Sorting modalities have been developed to achieve high enrichment factors, recovery rates, throughput, and purity. The negative downstream clinical impacts of sorting mechanisms using labels led groups to forego this option whenever possible in favor of label-free methodologies. No device has yet to produce preferred results for all metrics, but some, especially filters, can realize several metrics. Dead-end filters, specifically, are known for exceptionally large enrichment factors, purity, and throughput at the cost of recovery percentage or yield. Enabling filtration tools through novel flow fields to eliminate or minimize these losses can impact the diagnostic and therapeutic markets.

The literature suggests that rapid changes in flow rate have the largest effects on membrane clearance[1–4]. Pulse Width Modulation (PWM), a technique for controlling the proportionality of high to low signal (duty cycle), Pulse Amplitude Modulation (PAM), a technique for controlling the peak amplitude, and periodicity or frequency shifting of a square wave controlling volumetric flow rate and transmembrane pressure are shown to be novel techniques to increase recovery percentage in dead-end filtration systems while minimizing throughput tradeoffs. The objective of this thesis was therefore to employ pulse modulation to periodically backflush dead-end filters during sample processing for biological and non-biological particulate suspensions. The central hypothesis was that PWM flow control can interrupt cake formation through rapid rate of change of flow, reintegrate fouling layers into the bulk of a sample, and improve permeate flux. We

hypothesize that we can maximize recovery percentage by controlling the proportion of Q(t) through V_f, V_r, ϕ, α, and β. Further we hypothesize we can minimize processing time while still disrupting cake formation through control of flow rate proportions and frequency.

1.1 Specific Aim 1: Modeling and Validating Pulse Width Modulation Control to Improve Recovery Percentage.

Hypothesis: PWM periodic backflush of suspension fluid velocity enhances the recovery percentage of dead-end filters by interrupting cake formation, reintegrating the fouling layer into the bulk flow, and reinitializing processing. A mathematical model was developed to explore the factors influencing PWM backflush. The model incorporates a binary separation dynamic with 100% rejection of particles larger than the defined pore size. The model calculates flux across the membrane as a function of time and fouling layer formation, particulate permeation capacity of the filter, changes in bulk concentration, and filter fouling dynamics. The model is used to determine the recovery percentage as a function of duty cycle for fixed back and fixed forward flow (FBF & FFF) volumes. The model is validated experimentally by testing the filtration of polystyrene microspheres of 2.19 and 7.32 μm diameter. Microspheres were filtered through 5 μm pore size filters to experimentally determine gains in recovery percentage, enrichment factor, concentration fold change, and throughput in relation to the duty cycle of PWM-controlled flow.

1.2 Specific Aim 2: Optimize Pulse Modulated Backflush to Maintain Throughput and Recovery.

Hypothesis: Pulse amplitude and frequency affect the processing time and clearance of particulate from fouling layers. Modulation of periodic backflush waveform frequency and amplitude are used to minimize the reduction of throughput observed in PWM backflush. Two conditions were tested that explore the impact of frequency and amplitude on recovery percentage and throughput. The amplitude of the wave correlates to the volumetric flow rate. We determined that changing the amplitude of the negative flow rate while maintaining the backflush volume exchanged reduced processing time by 17.1% of FBF processing time while maintaining recovery percentage. The system uses time compensators between each flow direction change. We show high frequency waves can use smaller compensators and small errors in discrete approximations of higher frequency duty cycle can be leveraged to minimize processing time. An investigation was made to find the combination of effects which maximize throughput.

1.3 Specific Aim 3: Apply Pulse Modulated Backflush to cellular and non-cellular systems.

Hypothesis: PM periodic backflush can be used to improve the sorting efficiencies and recovery percentage in real-world applications, including conjugated microspheres and Cystic Fibrosis (CF) model bacteria. Moreover, a third application was tested in which perfusion bioreactor scaffold seeding was improved with PM flow control in both cell

uniformity and density. Protein-conjugated silica microspheres were created and isolated using PM techniques to investigate recovery percentage effects in a process relevant to industry therapeutics manufacturing. To simulate a sample from a CF patient, a disease analog, a549 lung epithelia and Staphylococci Epidermidis mixed, was created and the bacteria isolated using the PM technique. Sorting efficiency in comparisons to standard filtration was shown to be 2.1x larger. A perfusion bioreactor system was also tested to seed extracellular matrices and collagen sponges with microspheres and cells. Micro-CT and software analysis were used to measure seeding uniformity and density in comparison to standard perfusion. From practical systems, we can establish optimal conditions for application of the flow profiles.

1.4 Innovation and Significance

Pulse modulated periodic backflush is a useful and innovative approach to controlling fluid flow rate that contributes significantly to the advancement and revitalization of dead-end filtration systems. Dead-end systems with PM backflush are able to outperform cross-flow filtration devices in both recovery percentage and throughput. This thesis work initially investigates PWM theoretically and experimentally (SA1) as a proof-of-concept for enabling the use of dead-end systems to isolate target particles. PWM backflush was further tested through modulation of pulse amplitude and frequency to minimize reductions to throughput (SA2). The PWM-controlled flow concept was then applied to better isolate conjugated microspheres and bacteria and improve perfusion

bioreactor scaffold seeding uniformity (SA3). The concepts and process implementation are innovative because current dead-end systems show low recovery rates and rapid loss of permeate flux, Figure 1. In our system, we show an 18-fold improvement in particle recovery percentage and a 7-fold improvement in throughput compared to optimized cross-flow systems. This improvement is significant because we show it can enable the use of dead-end systems with improved recovery percentage and throughput for three applications in particle purification and cell and tissue engineering, expanding the capabilities of dead-end systems.

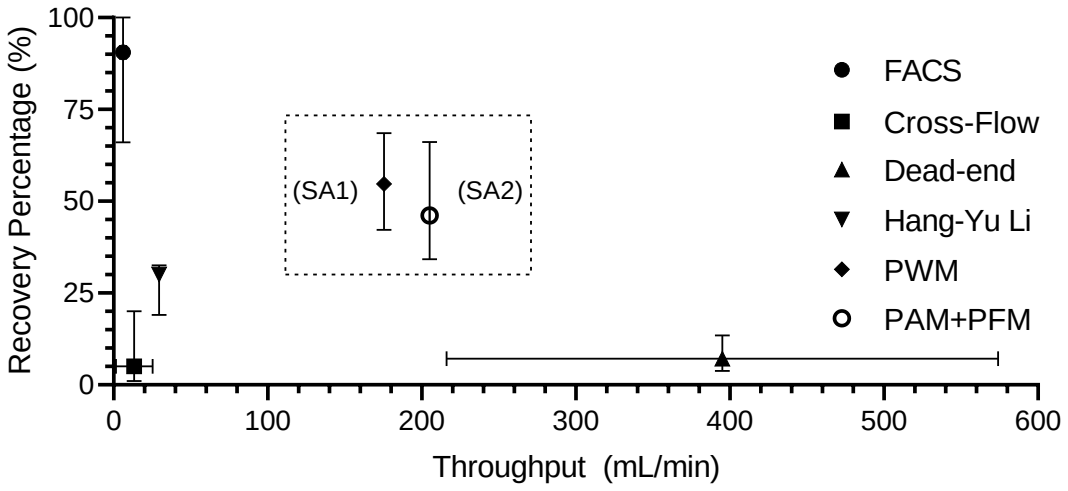


Figure 1 – Scatter plot showing current techniques used in filtration for recovery of suspended materials: Recovery Percentage vs. Throughput. Top right corner of graph is ideal, generating 100% recovery as fast as possible. Plot shows industry standard techniques median and range: FACS, Cross-flow, and Dead-end filtration; Cross flow improvement by Hang-yu Li's waveform modulation of cross-flow systems, and the impact of pulse width modulation (SA1) and pulse amplitude modulation and pulse frequency modulation (SA2).[1,4–8]

CHAPTER 2. BACKGROUND

2.1 Sorting

Sorting is the process by which a targeted material or item is selectively moved or removed from a population of materials or items. Many biological testing processes, ranging from therapeutic purification to cellular isolation, rely on sorting principles. Further, extracted biofluid composition is often heterogeneous, and depending on the source, can possess a mixture of cell types and biomolecules that can interfere with analysis[9]. As end users, we primarily focus on the inputs and outputs of sorting protocols, treating the steps in between as a black box, but understanding the types, applications, and results of sorting is vital for applying appropriate techniques. These techniques for sorting can be separated into two main methodologies: labeled and label-free.

2.1.1 Labeled Modalities

Labeled sorting modalities rely on manipulation of the target particle and/or sorting device to achieve separations. These manipulations generally rely on the attachment of biomolecules to the surface of the targeted particle in suspension, predominately using antibodies labeled with an identifier and tailored to bind to a specific substrate. Three primary labeled techniques fluorescently activated cell sorting (FACS), magnetically activated cell sorting (MACS), and panning are commonly used for isolation of targeted cellular substances or microparticles.[9]

Labeled techniques can generally provide incredibly large purity, enrichment factors, and recovery percentages, but this comes at the cost of throughput. Further, most labeled techniques will affect cell fate or function, and can be contraindicated for downstream analysis and/or the efficacy of therapies. Choice of antibody may be limited, hard to manufacture or isolate, and incredibly labor intensive or time consuming to identify, purify, and acquire, limiting labeled processes to a few approved techniques or research purposes only. These shortcomings have led to a resurgence of label-free sorting modalities in the diagnostic and therapeutic space.

2.1.1.1 Fluorescently Activated Cell Sorting

FACS is an industry standard technique that utilizes monoclonal antibodies modified with attached fluorophores or fluorescent micro- or nanoparticles. This technique, and other antibody-based solutions, are very specific, attributable to the high specificity of the antigen antibody pair selected for targeting. Once conjugated, incubated, and adhered to the target particle, samples can be fed through a flow cytometer based fluorescently activated cell sorter to detect fluorescence and selectively remove fluorescent particles from the heterogeneous population. Optical sorting techniques can offer throughputs in the range of approximately 10,000 cells/s, which is large but relatively low compared to alternate bulk sorting methods whether labeled or label-free because cells must be isolated independently. Further, the FACS approach is one of the most expensive options, requiring a large flow cytometer to process the cell suspensions[10].

2.1.1.2 Magnetically Activated Cell Sorting

Similar to FACS, MACS is an antibody-based sorting technique. Utilizing targeted antibodies, researchers and clinicians can conjugate magnetic micro- or nanospheres to antigen present cells or particles. These structures can then, in bulk or in a flow channel, be exposed to a permanent or electromagnetically generated magnetic field, affecting the movement of the conjugated spheres and, by extension, all things bound to them. This technique offers a much higher throughput in isolation compared to FACS because the separation can be conducted in bulk samples, but higher throughput in MACS often comes at the cost of purity as it is difficult to gauge and control movement based on antigen expression, providing an all-or-nothing type response, ignoring antibody saturation[11,12].

2.1.1.3 Panning

Panning is the process of selectively pulling cells out of suspension by adsorbing antibodies to the surface of a device or container and binding out the targeted cells. Panning requires a lot of preparation and is not very repeatable for multiple samples, usually requiring a new device or container for each cell suspension. Induction of binding is also a problem. To encourage cell adhesion, several groups have built microfluidic devices specifically designed to force surface interactions or induce cell rolling over antibody coated surfaces. Using panning systems will generally produce high purity samples of targeted particulate through collection, but throughput and recovery percentage are locked

in a balancing act. Increasing throughput on the device means sacrificing recovery percentage and vice versa[9].

2.1.2 Label-free Modalities

Label-free sorting methodologies rely on the inherent physical properties of cells or particulate to isolate them. These inherent properties can include cell density, size, stiffness, charge, adhesiveness, polarizability. Not all of these properties are inherent to or differentiable between all cell types, and multiple techniques may need to be employed to achieve desired results. Almost all label-free techniques enable bulk processing of cellular samples and, perhaps more importantly, produce suspensions that are immediately available for downstream analysis or therapeutic use by limiting form and functionality manipulations.

2.1.2.1 Centrifugation

Centrifugation, or sedimentation, is a density-based label-free sorting technique used almost universally. Variations in density of suspended particulate in relation to other particles and the suspension fluid results in floating or sinking over time. Higher density particulate will have a higher average settling velocity compared to lower density particles in a given suspension. Settling velocity disparities lead particles to collect in layered groupings at the bottom of the suspension vessel. This can be expedited using a centrifuge. A centrifuge is a rotor-based device that spins fluid based samples, generating a centrifugal

effect, forcing particulate to settle faster as though exposed to many times the gravitational pull of earth. Once settled, each layer can be removed, or samples from each layer can be selected. Density-based isolation is particularly useful in blood separations as centrifugation will create three distinct layers: red blood cell (RBC), buffy coat (containing white blood cells WBCs), and plasma and proteins. Scientists can then aspirate off sections for analysis and use. Although quick and useful for bulk separations, this process fails to yield high purity samples without sacrificing recovery percentage[13–15].

2.1.2.2 Microfluidics

With the modern advancement of fluidic device production techniques, many microfluidic devices have recently been developed to sort cell content based solely on label-free modalities to achieve high purity, enrichment factor, recovery, and throughput[16–21]. There are many reasons for pursuing microfluidic based technologies: the scale of the systems allows for the manipulation of single cells, the volume of sample required can be cut allowing for point-of-care diagnostics or processing of samples for multiple diagnostics from a limited source, and the portability is much higher and peripheral technology cost much lower (some devices operating solely via syringe pump). However, most microfluidic devices operate in regimes governed by non-intuitive physics. Many devices utilize laminar flow and high fluid velocities or viscosities to achieve tailored results. Device variation enables a wide range of sorting methods including acoustophoresis, deterministic lateral displacement, microstructure trapping, inertial differentiation, dielectrophoretic, and

optical among others. These techniques offer a huge array of very specific options to isolate and enrich cells and particles from a suspension, but manufacturing can be very difficult to scale-up, cost can be high, and the limited size severely effects throughput and bulk processing[5].

2.1.2.3 Filtration

Filtration is label-free technique that has been in use for thousands of years. Numerous publications exist primarily focused on filter use in water, beer, and milk-based studies, especially when looking at biofouling. Filtration is a size and stiffness-based sorting modality that uses exclusion and pressure differentials to eliminate and purify content. Membrane based systems will have pore diameters either greater than or less than the diameter of the targeted particulate. Pore diameters provide a binary sorting modality that keeps particulate larger in diameter than the pores as retentate and allows the particulate smaller in diameter to pass through as permeate. Particulate deformability also plays a role in these types of separations as particles larger than the filter pore size may pass with sufficient flow pressure. A poignant example of the role particulate plays is the filtration of red blood cells. Although RBCs have an average height and diameter of approximately 2 μm and 7 μm respectively, these cells are incredibly pliable and can deform to a critical diameter of about 2.5 μm regardless of orientation[22]. When utilizing filters, particle deformability informs selecting pore size.

Further, filtration can be used on a macroscopic scale, for bulk processing large volumes, or on a microfluidic scale, integrated as pillars or traps in a microfluidic device[2–4,22–25]. New developments and technologies have revitalized filtration as a sorting modality. It has long been a favorite sorting technique for clinical separations, and with new research developments, it offers some of the greatest advantages when quantifying successful filtration operations. It is for these reasons that we are looking deeper into filtration as a sorting technique with the goal of improving particulate flux and reducing the negatives associated with some filtration modalities.

Unfortunately, challenges associated with label-free filtration include need of equipment that is expensive and has a large footprint. For example, industrial cross-flow filtration units can be costly, only specifically effective, and utilize tens to thousands of square meters of filtration media and the priming fluids and reagents to keep it functional. Size exclusion, however, is a well-known thoroughly explored field that avoids many of the negative affects driving the push toward a label-free sorting dominance and is currently undergoing a renaissance as new tools and techniques become available.

2.2 The Need for Improved Enrichment and Recovery of Cells and Particles

Isolation through filtration of desired or target particles and removal of undesired, untargeted particles is a standard technique used by laboratories, hospitals, and industrial settings. The sorting process is used in a wide range of spaces from disease diagnostics, to sample purifications for therapeutics, to cellular isolations for study, and in treatment of

cell and tissue injury[10,26–29]. Each of these applications have many sorting or isolation protocols employed. Each protocol should improve outcomes of a procedure, such as reducing misdiagnosis, adverse treatment effects, or therapy failures. As a result, the world of separations is well explored and focused on achieving high enrichment and recovery percentages while mitigating negative consequences to achieve successful results.

High enrichment and recovery percentage sorting techniques have two primary modalities: labeled and label-free[9,18]. Whenever possible, the clinical sorting market has moved away from labeled techniques. Labeled processes can be contraindicated for the downstream use after sorting, preventing their use in clinical settings, and they tend to be hampered by several limitations including large instrumentation, low throughput, cell death, limited quantitation, intensive regulation, and high costs[9]. Unfortunately, some label-free techniques can suffer from the same problems. For example, in order to produce large recovery percentages and throughputs in cross-flow filtration systems, industrial manufacturers need to build massive processing plants with up to hundreds of meters of piping systems[4,23,30–32]. However, dead-end size exclusion avoids many of the negative effects with generally low costs and a wide range of available instrumentation footprints.

Label-free, size-based, filtration of suspensions is currently used in particle separations, cellular isolations, sample purification, bioreactor systems, and cell and tissue engineering. However, dead-end processing is susceptible to membrane fouling and failure, resulting in low recovery percentages for samples. As a result, many clinical and

industrial groups have been limited to low throughput cross-flow systems which allow for much higher recovery via recirculation of retentate. Improving dead-end filtration to increase recovery percentage and throughput would greatly improve the efficiency of its use for disease detection and may revitalize the technology for use in the medical field.

2.2.1 Measuring Filtration Success

Filtration is a well-established field, with methods dating back to 4000 BCE[2–4]. With modern advancements in analysis and filter creation techniques, a more thorough understanding of filter performance can be conducted[23,33–35]. When used in separations or isolations in dead-end systems, any material that passes through the filter is known as permeate and anything retained by the filter is identified as retentate. The field uses four main metrics to analyze filtrate when isolating targeted from untargeted particulate: Purity[36,37], Enrichment Factor (EF)[19,38], Recovery Percentage[28,39,40], and Throughput[37,41,42].

$$\text{Purity} = \frac{targetted\ particle\ count|_{Output}}{total\ particle\ count|_{Output}} \tag{1}$$

$$\text{Enrichment Factor} = \frac{\left(\frac{targetted\ particle\ count}{nontargetted\ particle\ count}\right)_{Input}}{\left(\frac{targetted\ particle\ count}{nontargetted\ particle\ count}\right)_{output}} \tag{2}$$

$$\text{Recovery Percentage} = \frac{targetted\ particle\ count|_{Output}}{targetted\ particle\ count|_{Input}} \tag{3}$$

$$\text{Throughput} = \frac{Recovery\ Percentage}{Total\ Processing\ Time} \quad (4)$$

Purity measures the percentage of targeted particulate matter in the output of a system compared to total particulate in that system. Enrichment factor is a ratio of ratios that shows the proportion of targeted to nontargeted cells in the output compared to the initial sample being processed at the input. Enrichment factor ranges from 0 to infinity where a value of 1 implies the permeate contains the same proportion of targeted to untargeted particulate as what was supplied producing no isolation. A value of infinity would result from perfect permeate purity without any contamination. Enrichment factor does not indicate an input or permeate concentration or quantity; this is determined by recovery percentage. Recovery percentage provides the number of targeted particles at the output compared to what was initially supplied at the input. Throughput is a measure of the output quantity as a function of processing time. Different filter types and processing methods will trade off performance in one of these metrics for performance in another. Table 1, below, provides the equations for calculating these metrics and shows how different filtration modalities will generally perform compared to each other.

Filter	Purity	Cell Enrichment Factor	Recovery %	Throughput
Dead-end	Highest	Highest	Lowest	High
Cross Flow	Highest	High	Varies	low
Microfluidic	Varies	Varies	High	Lowest
Weir	High	High	Low	Low
Equation	$P = \left(\frac{targeted}{total_{output}}\right)$	$EF = \frac{\left(\frac{targeted}{nontargetted}\right)_{output}}{\left(\frac{targeted}{nontargetted}\right)_{input}}$	$R\% = \left(\frac{targeted_{output}}{targeted_{input}}\right)$	$T = \left(\frac{Recovery}{Process\ time}\right)$

Table 1 – Gap analysis of the four metrics used to quantitatively measure filtration. Table compares recovery percentage, enrichment factor, purity, and throughput between four different systems used to filter particles and cell content and provides equations for calculating each respective metric.

2.2.2 Microparticles in Therapeutics Research and Processing

Microparticles, especially those engineered for stimulus response, have been used ubiquitously for biomedical purposes. These may include environmental, intracellular or pathology-associated triggers such as pH[43], target enzymes[44], temperature[45], and redox potential[46]. Other stimuli may be externally-applied, such as magnetic fields[47], light[48], and ultrasound[49], or may employ stimuli in combination[46,50]. Such responsiveness has allowed for burgeoning development of therapeutic and diagnostic biomaterials of ever-increasing precision. Even non-degradable microparticles are useful *in vitro* to study biological phenomena. Pacheco *et al*, for example, demonstrated that for IgG-functionalized polystyrene particles, classical complement pathway activation increases commensurate with Fc density and inversely with particle size[51], while macrophage uptake for smaller Fc-

coated particles (1 μm or less) was found to be related to the extent of IgG coverage [52]. These insights have informed the design of vehicles to develop host-modulated antibacterial approaches [53] and to maximize bioactive payload delivery to macrophages and/or tune their downstream activity.

Additionally, microparticles are used in therapeutics processing as well. There are numerous techniques that make up a multibillion dollar industry for separations in pharmaceuticals and biotechnology[54,55]. These techniques are focused on the isolation of cellular components[56–58] through adsorption to functionalized particles or antibody collection on the surface of microparticles directly[59,60]. To achieve separation and collection with microparticles, the microparticles are added to suspensions or columns, incubated to allow time for adsorption and binding, filtered, and the collected protein therapeutic released.

One of the main issues in drug therapy development is scaling. For a microparticle therapy to be useful, it must be effective in its use case and scalable to treat the disease nation- or worldwide. Many production techniques will utilize size-exclusion processing techniques to achieve high purity within their samples for validation purposes at the investigational stage that require highly pure, but relatively small, microparticle sample quantities. In-vitro, and even some in-vivo studies, do not require large doses of microparticles to demonstrate effectiveness of the approach. To increase scale, many experimentalists will utilize cross-flow processing techniques to acquire the necessary

purity while avoiding membrane fouling, but throughput for cross-flow filtration is low when scaled, and dead-end filtration of large sample quantities requires filters with hundreds of square meters in effective filtration surface area to prevent loss associated with fouling and flux decay.

2.2.3 Biofouling and Biological Separations

Bacterial infection continues to be a leading cause of death in both developed and developing countries, and with the rising frequency of antibiotic resistant infectious agents, lack of discovery of new antibiotics, and rising diagnostic capabilities of clinics, broad spectrum antibiotic based treatments are quickly being removed from the standard of care[61,62]. This is especially true for diseases that result in frequent, recurring bacterial infections.

A particular challenge of bacterial isolation is, because biological samples contain long chain proteins that easily adhere to membrane surfaces, dead-end filtration systems will foul faster than under normal particulate filtration processes[61,63]. This fouling increases the time needed to complete the cell culture step of the care pathway by producing small colony seeding densities. To identify pathogenic bacteria in BAL fluid, the bacterial agent must be isolated and then cultured for enough doubling times to produce sufficient cellular mass for analysis. Cell culturing techniques currently takes anywhere from 24 to 48 hours to culture the bacteria to a critical mass, and is the rate limiting step for infectious agent diagnosis for targeted therapy. By increasing recovery percentage in the filtration stage of

the care pathway, it may be possible to minimize false negative rates for infection diagnostics and reduce the care pathway rate limiting step to achieve same day diagnostics.

2.2.4 Perfusion-based Bioreactors: Seeding Density and Uniformity

In the orthopedic tissue engineering field, large deficits resulting from trauma, surgery, or slow or non-healing defects can be found in bone, cartilage, tendon, and ligament tissues. The more than one million orthopedic repair surgeries per annum costs the U.S. more than $5 billion[64,65]. Many of these defects are repaired through allograft, autograft, or xenograft where new tissue is sourced from a less critical tissue and transplanted to replace the defect[64]. However, problems such as rejection, graft versus host disease, autoimmune destruction, and limited material sourcing prevent optimal or even complete repair[66]. These problems have led to investigations into bioreactor based tissue engineering solutions in which replacement tissues are engineered from host cells combined with either decellularized scaffold materials or artificially constructed replacements[67–71].

Bioreactors are systems designed to maintain and developing biological reactions. Reactors generally consist of a feeding pump, reactor tank, feedback sensor probes, agitators, and effluent pump. The main types of bioreactor are rotating wall vessel, spinner flask, compression system, and perfusion bioreactor[71]. Each of these systems has their own shape and function, but all are used to culture cells. The perfusion and compression systems

both incorporate a decellularized or nonbiological scaffold as the foundation on which cells can be grown.

These scaffolds can come in many shapes and sizes depending on application, but they are primarily categorized by composition: monolithic, bi-layered, multi-layered, or gradient[65]. The scaffolds can have tunable biodegradability, cellular conductivity, porosity, and mechanical properties based on the material used to construct the scaffold. In general, however, all scaffolds behave as porous membrane on and through which cells are seeded to grow new tissues.

We believe decellularized scaffolds, when loaded into perfusion-based bioreactor systems, behave akin to dead-end micro-sieve filters, capturing and retaining perfused cells and debris on the surface and top interior of the scaffold. Static seeding often yields low seeding efficiencies and poor cell distributions; thus, creating a need for techniques that can improve these parameters[72]. If the scaffolds behave like dead-end filtration devices, they are susceptible to the same problems of fouling and cake formation that standard filters suffer from, preventing high density seeding through large scaffolds or sacrificing uniformity throughout the scaffold[72].

$$\text{Percent Uniformity} = 100 * \left(1 - \frac{\sigma}{\bar{x}}\right) \tag{5}$$

As a result, groups have explored new or varied seeding protocols to achieve the desired scaffold seeding such as inversion, high pressure, and oscillatory flow. Improving the

recovery percentage of these types of filters may lead to reduced fouling and better permeation, seed density, and uniformity[72].

2.3 Understanding Filtration

2.3.1 Membrane Types

Filtration of larger volumes operating on a scale for bulk sorting primarily utilize membrane-based filters[1–4,8,23,73]. Membranes are integrated into the fluidic circuit performing the filtration and comes in two main varieties: fibrous and microsieve. Fibrous membranes are constructed of tightly woven fibers resulting in a distributed pore diameter with higher inter-pore variability[40]. The cross hatching of fibers in multiple directions ends up restricting particulate by creating intra-membrane pores in the inherent tortuosity of the fibers as opposed to holes running through the membrane. Materials for these types of membranes are primarily cellulose based with specific coatings to affect the hydrophobicity of the membrane. Fibrous membranes are used frequently in cell sorting because the volumetric flux of the suspension fluid is separate from that of the particulate flux. Because caking and pore blocking generally do not affect smaller cross-hatched pores on the membrane surface, suspension fluids can still flow through the membrane even when the larger pores are blocked[74].

Patterned membranes, or microsieves, operate differently than fibrous membranes in that solvent flux is directly linked to particulate flux through membrane resistance and

fouling[3,23,40]. These filters, as the name implies, have structured pores patterned across the surface of the membrane running through to the other side. Pores are usually cylindrical in shape but can be tapered or specifically designed to have triangular or square cross sections. Volumetric suspension fluid flux and particulate flux are tied through membrane resistance for these filters. As pores become blocked paths for fluid flow become limited and membrane resistance is driven up. Membrane resistance requires closer monitoring for cellular samples because higher pressures can result in cell shearing as they are being sorted. Patterned filters, however, have a very tightly distributed pore diameter, and manufacturing controls can be well designed for specific applications, making them an ideal choice for sorting rigid particles.

The experiments run in specific aim 1 were validated with several membrane filters, but the data presented herein stems from experiments utilizing commercially available Pall Acrodisc syringe filters with Versapor and GE Whatman Nuclepore track-etched microsieve filters. These filters are composed of a hydrophilic polypropylene and polycarbonate membrane respectively. They are commercially used for both aqueous and organic samples with a glass-fiber prefiltration component on the Acrodisc filter. All filters have a diameter of 25 mm and functional cross-sectional area of approximately 2.8 cm^2.

Figure 2 presents the three modalities of sorting using filters: Cross-Flow, Weir, and Dead-End. Each of these arrangements of filters in a fluidic circuit has its own advantages

and disadvantages associated with the orientation of the bulk flow with respect to the exclusion modality.

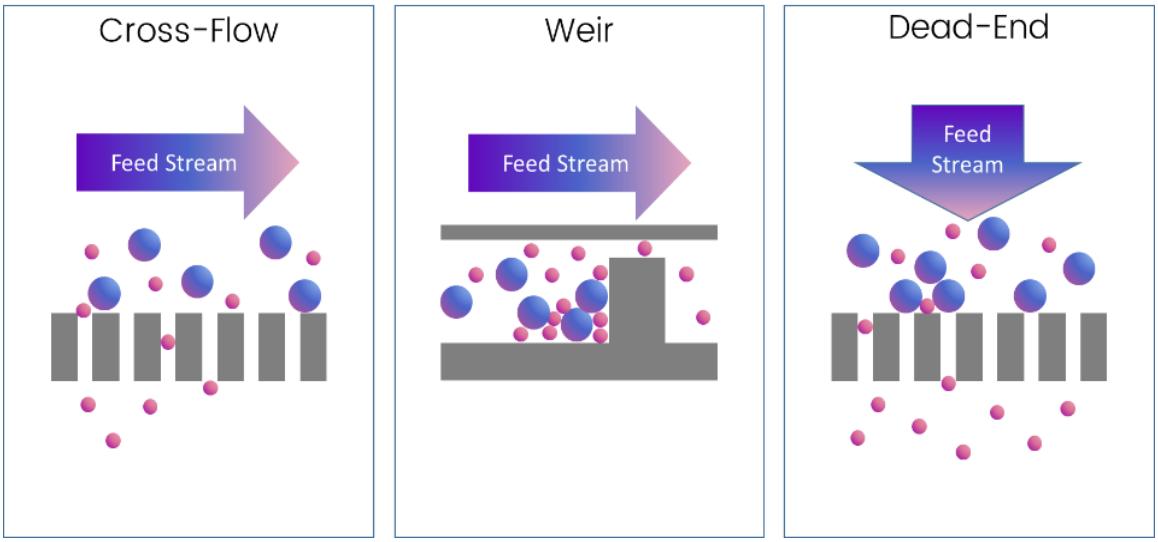


Figure 2 – Filtration types, A) Cross-flow filtration flow profile parallel to filter membrane in which small particles transverse a filter driven by a pressure or chemical gradient. B) Weir type includes traps and is mainly in microfluidic systems. C) Dead-end filtration flow profile perpendicular to filter membrane.

2.3.2 Crossflow Filtration

Cross-Flow filtration, one of the primary modes of filtration for its high recovery, is a technique wherein the bulk flow from the feed stream runs parallel to the filtration surface[1,2,22]. Filtration in a cross-flow orientation significantly decreases the throughput and recovery percentage compared to other membrane-based filtration techniques because it is dependent on particle migration. The gradient across the filter leads to net migration of targeted particles through the membrane while any buildup of excluded particles tends

to be swept away by the shear effects of the feed stream at the membrane surface. To overcome the losses associated with cross-flow filtration, any retentate in the feed stream is usually recirculated to process again, decreasing throughput further.

2.3.3 Weir-Type Filtration

Weir-type filtration is a filtration technique usually reserved for microfluidic systems. It involves the construction of cell traps specifically designed to capture individual cells rather than individual types of cells, but can be integrated with inertial or gravitation systems to remove specific types of cells in bulk[19,22,24,37,75,76]. These filtration systems are usually limited to single use and have difficult or time intensive manufacturing processes. These filters are difficult to scale, but consistently produce ultra-high purities when enriching for rare cell types. Recovery percentage and enrichment also tend to be high (>80% and in the tens to hundreds respectively) in these devices, but only because the target populations used are particularly small and recovery of a few cells is all that is required to boast successful results.

2.3.4 Dead End Filtration

Dead-end systems are fluidic circuits in which the membrane filter is perpendicular to the feed stream. As a result, all particulate in the bulk sample must interact with the filtration membrane to complete flow through the circuit. The main challenge of dead-end filtration is the constant pressure differentials leads to decay in permeate flux with a

buildup of fouling material. To overcome flux decay, an increasing pressure is needed to drive a solute permeation. Dead-end membranes are particularly susceptible to membrane fouling and failure given the constant flux regime, leaving the systems to process most content in with constant pressure.

Dead-end systems are an ideal target for improvement of filtration systems. They have enormous enrichment factors (in the hundreds to thousands), high throughput (depending on surface area and concentration of bulk), and high purity. The primary metric in which dead-end filters fall short is recovery percentage, Table 1. Low recovery percentage is primarily attributable to fouling of the membrane pores, increasing membrane resistance.

2.3.5 Filter Fouling and Flux Decay

There are three primary modes that contribute to the fouling and flux decay of conventional, constant pressure dead-end systems: deep bed fouling, pore blocking, and cake formation[77–80]. Deep bed fouling occurs when particulate or debris adsorbs or becomes embedded or trapped in the deeper parts of the membrane or supporting material. This form of fouling is usually irreversible and preventative measures work best to eliminate it. Further, this modality will contribute rapidly early in membrane fouling, but eventually the support material will saturate with debris only increasing membrane resistance slightly as is observable in Figure 3.

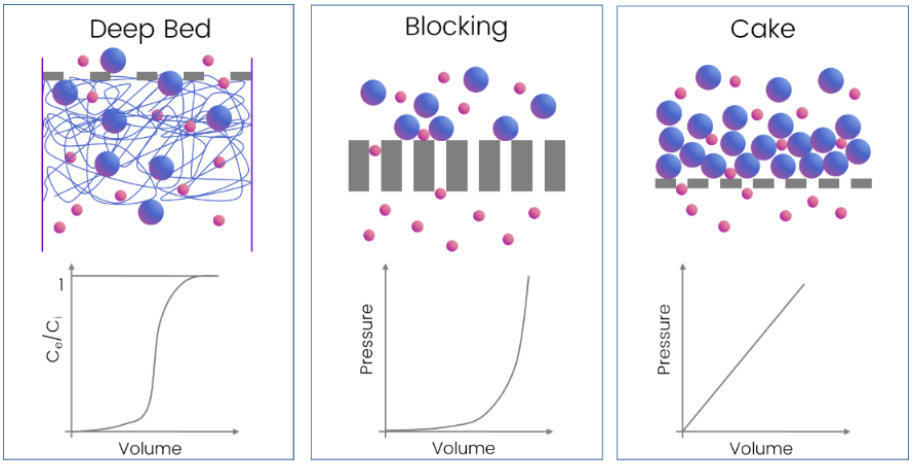


Figure 3 – A Depiction of A) deep bed fouling with associated plot of particulate efflux to influx saturating at 1, B) blocking with exponential growth of pressure, and C) cake layer formation with linear profile of pressure buildup.

Blocking and cake layer formation contribute more to filter fouling than deep bed. As can be seen in Figure 4, for microsieve based membranes, blocking increases membrane resistance and associated transmembrane pressure profiles through direct occlusion of pores. As pores become blocked, the number of pathways through which permeate can pass through the membrane exponentially decays. The buildup of additional particles on the blocking layer then results in cake formation. Caking contributes linearly to the growth of membrane resistance and overall decay in permeate flux[77]. Further, the longer these two forms of fouling interact with the filtration membrane, the more irreversible these interactions become[77,79,81–83]. We hypothesize that, periodic and frequent disturbance of these forms of fouling will greatly improve a membrane's useful lifespan and effectiveness. These three types of fouling combine to produce an exponential decay in the permeate flux

through a standard dead-end filter. The relationship between permeate flux and time can be described by three distinct phases: rapid decline, tapering, and stagnation. In typical dead-end systems, the permeate flux will asymptotically approach a steady state flux as cake builds.

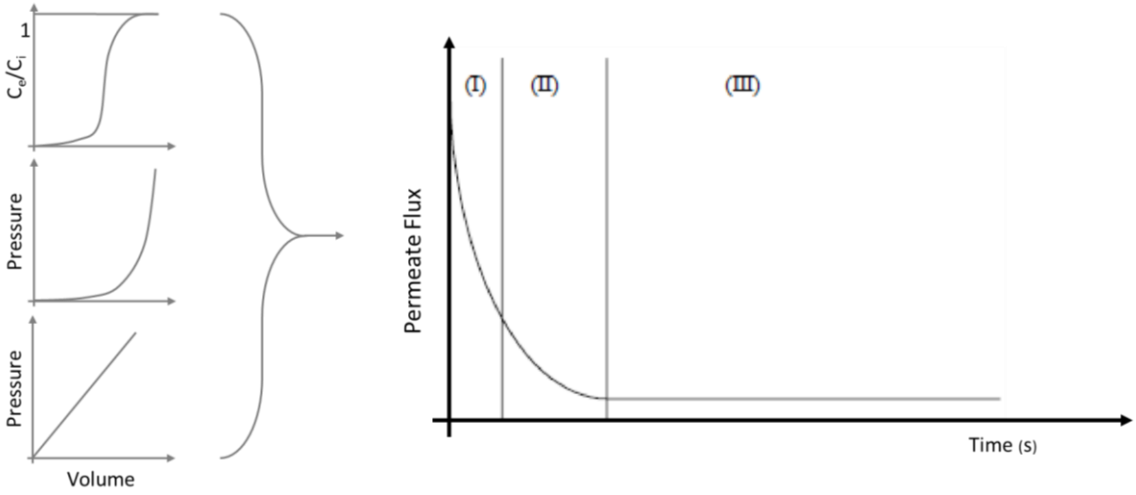


Figure 4 – Combination of three fouling effects resulting in exponential decay of Permeate Flux separated into three distinct phases: (I) rapid decline, (II) tapering, and (III) stagnation for continuous pressure dead-end membranes

The permeate flux can be approximated by an exponential decay equation[78]:

$$J(t) = J_0 * e^{-f(R_m)t} + J_{s.s.} \quad (6)$$

2.4 The Market for Biomedical Sorting Technologies

Biomedical sorting technologies can be divided across a few markets: laboratory automation, split into liquid handling and sample preparation, bio-specimen preparation,

and cell and tissue characterization and separation. The laboratory automation market spans clinical diagnostics, drug discovery, food and agricultural research, and forensics, and is quickly defining itself as a specific part of the laboratory industry in the U.S. In this market, "there is a vital need to reduce laboratory operating costs due to budget pressures throughout the laboratory industry and new applications of laboratory automation technology coming on board, so a great and broad-based demand for laboratory automation products will be evident during the next decade[84]." Further, the cell and tissue analysis market has observed an increasing quantity of novel tools and techniques targeting cancers, autoimmune disorders, and tissue engineered implants[85]. The rapid growth in these sectors is centred around novel techniques to improve processing and cut costs.

Combined, sample preparation and liquid handling accounted for $1.21 billion in revenue in 2012 with compound annual growth rates (CAGR) of 5.8% and 6.5% respectively, and was projected to have total revenue of $1.68 billion by 2017[84]. Further, the bio-specimen preparation, and cell and tissue characterization and separation accounted for $7.12 billion in revenue in 2012 with compound annual growth rates of 6.2% and a projected total revenue of $9.63 billion by 2017[85], providing a total addressable market of $11.31 billion with an average CAGR of 6.17%.

2.4.1 Market Positioning

Academic laboratories will drive growth in the liquid handling market through the purchase of low-cost systems[84,85]. Even though spending from academic institutions will

decrease in the broader definition of laboratory automation, new applications of automation within the clinical space, including point of care diagnostics and clinical sample preparation were main drivers of growth through 2017[84]. We can track academic institution output through publications. Shields et al. shows that the number of publications about separations in the microfluidic space, from the years 2000 to 2015, followed an exponential growth curve. However, they also indicate that medical and basic science applications only make up only about 27% of total publication content with medical applications making up about 11% of overall publications[5,86]. Further, Shields discusses the value of label-free sorting and filtration for the minimal preparation required and few contraindications for downstream applications. We investigated the number of publications in the filtration space, Figure 5. We found similar quantities of publications in a similar proportion when investigating filtration but noticed a more linear trend in publications on larger scale systems and a surprising lack of literature on dead-end filtration and even fewer investigating flow control.

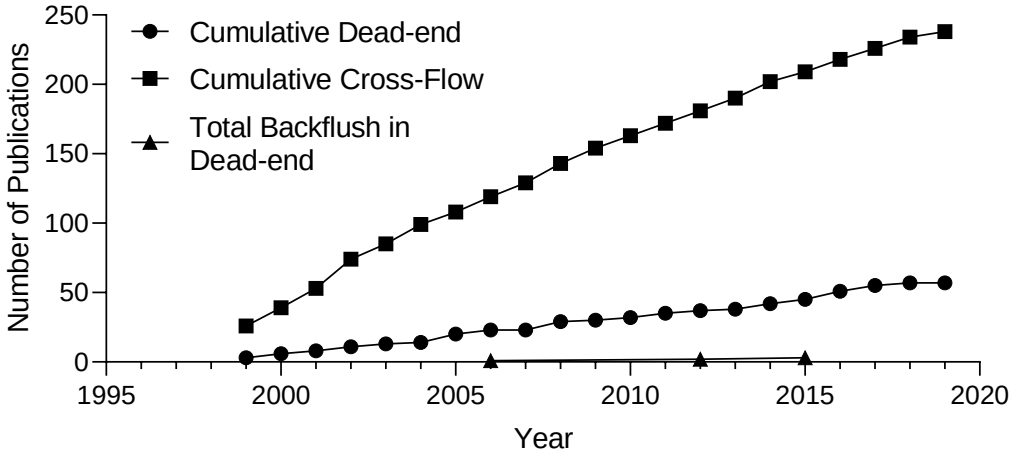


Figure 5 – Cumulative publications year over year for dead-end and crossflow filtration.

There is a strong set of literature investigating optimization of backflush. The literature heavily explores minimizing backflush volume to improve cake resistance reduction percentage, little investigation was conducted outside of this space. In a plot of frequency vs duty cycle, Figure 6, you can very easily identify clustering around a duty cycle range of 0.85 to 1.00 and frequencies below 0.01[32,81–83,87–89]. The reviewed literature is primarily focused on minimizing backflush durations to improve solvent flux in large volume systems and not focused on recovery percentage of solutes. However, the investigation performed by Hong-Yu Li et al.[1], deviate from this cluster in the frequency range and show marked improvement in recovery percentage and throughput as a result, Figure 1. We theorized that exploration outside of the cluster could have dramatic implications on filtration effectiveness for the recovery of permeate particles. In this thesis,

we discuss in detail how the exploration of a full range of duty cycles and frequencies around the best performing ϕ can be used to maximize recovery percentage and throughput.

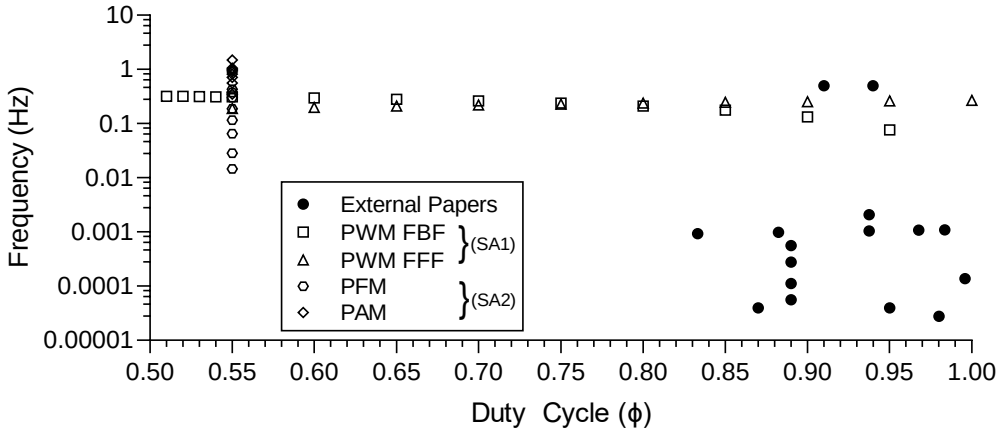


Figure 6 – Scatter plot of frequency vs duty cycle on a log scale for papers from the literature review compared to the range explored in SA1 and SA2. The higher frequency pair from the literature review indicate the best performing investigations of Hong-yu Li in improving cake resistance reduction percentage for different waveforms. PFM and PAM explorations were pursued at a duty cycle of 0.55, the best performing ϕ in terms of recovery percentage with the lowest throughput.

Three major companies, PerkinElmer, Eppendorf, and Gilson maintain a significant majority of the market share for liquid handling. However, the sample preparation market space is dominated by large conglomerates, including The Marmon Group, Roche, and Beckman Coulter, and the well established companies Graver Technologies, ThermoFisher, PerkinElmer, Qiagen and Tecan[84]. The structure of this industry gives us a small but competitive array of institutions with established distribution channels who are exploring this space and would be interested in the work presented herein. We have received interest in an intellectual property licensing deal from one of these firms, and

expect single subsidiaries of these major institutions, such as Graver Filtration Technologies, to express continued interest in this research because they would be able to quickly integrate the technology and make a significant impact in this space.

2.5 Discussion

The literature indicates a clear need for improved isolation and recovery of particles and cells. With recent advancement in technology, the market is open to novel techniques that help to move away from labeled modalities toward label-free options. Through a deep understanding of the types of membrane systems and what causes them to fail, we have identified a gap which revitalizes dead-end filtration systems and improves upon current practices. By establishing PM flow control, we can interrupt cake formation through rapid rate of change of flow, reintegrate fouling layers into the bulk of a sample, and improve permeate flux to optimize recovery percentage and throughput in dead-end systems.

CHAPTER 3. WAVEFORM MODULATION

3.1 Duty Cycle

To motivate our approach to solving the problem of membrane fouling, we consider that flow profiles can be manipulated through non-uniform pressure control in order to clear fouled membranes. To understand how, we redefine duty cycle, a term which usually refers to the time occupied by the cycle of operation of a machine or other device, as a function of volumes moved forward and backward through a membrane. This metric is used by mechanical and electrical engineers to define the percent of a period that a machine or waveform remains in the 'on' or functioning state compared to the total operating time in a recurring period. Pulse width modulation (PWM) is a technique used to vary the duty cycle of a waveform controlling an actuator to achieve a desired effect.

A few groups have tangentially investigated waveform control of fluid flow to minimize negative effects of backflush without looking directly at duty cycle[1,32,81–83,87–91]. The priority for many of these groups is to extend a filters useful life span, mostly in cross-flow apparatuses, while minimizing backflush volume to do so. This focus fails to investigate the impact of waveform control and clusters the findings around a duty cycle of 90% as a result. Only one group we've identified investigates waveform control directly by comparing the effects of multiple types of waves, but applies their research to cross-flow systems and does not look at intra-wave-type variation including periodicity, duty

cycle, or amplitude variation[1]. However, Hong-yu Li does show improved clearance of cross-flow systems compared to the other groups by direct waveform control.

We propose to increase permeate flux, enrichment factor, and particularly recovery percentage in dead-end systems by implementing a PWM periodic backflush fluid flow cycle to clear reversible fouling, blocking and cake, of the membrane. We define the ratio of forward flow volume V_f to backflush volume V_r during a filtration operation as the duty cycle, ϕ:

$$\phi = \frac{V_f}{V_f + V_r} \tag{7}$$

For a better understanding, a ϕ of 1, $V_f = V_f + V_r$, or $V_r = 0$, constitutes continuous forward flow with no backflush and is subject to standard exponential decay in flux as fouling occurs. Comparatively, $\phi = 0.50$, $V_f = (V_f + V_r)/2$, or $V_f = V_r$ has the solvent volume dedicated to forward flow through the membrane equal to the solvent volume for backflush in any given periodic cycle and no net processing would occur. Equation (7) can be expanded to include a time-based analysis where α is the time required to complete a volume of forward flow given the flow rate of the solvent through the membrane, Q(t), and β is the time required to complete a volume of backflush given a flow rate of the solvent through the membrane.

$$\alpha = \frac{V_f}{Q(t)} \; ; \; \beta = \frac{V_r}{Q(t)} \tag{8}$$

In Chapter 4.4, we discuss the effects on recovery percentage we observe as we vary the relationship between V_f and V_r. We hypothesize that we can maximize recovery percentage by controlling the proportion of Q(t) through V_f, V_r, ϕ, α, and β. Further we hypothesize we can minimize processing time while still disrupting cake formation through control of flow rate proportions and frequency.

3.2 Approximating the Behaviour of Flux

Utilizing this waveform control, we can apply a square profile to volumetric flow rate to change the transmembrane pressure and temporarily reduce fouling. As addressed by Hong-Yu et al.[1], rapid changes in fluid velocity including a negative transmembrane pressure result in high cake resistance reduction percentages. Square waves will have the highest change in volumetric flow rate of all non-constant flow profiles. We hypothesize that we can leverage square wave flow profiles to reduce cake resistance to improve dead-end filtration. We predict three effects from these methods: interruption of cake and fouling layer formation resulting in membrane flux capacity improvement, reintegration of the fouling material into the bulk concentrate, and re-initiation of filtration. For $\phi > 0.50$ the resulting transmembrane pressure and permeate flux profiles are described in Figure 7.

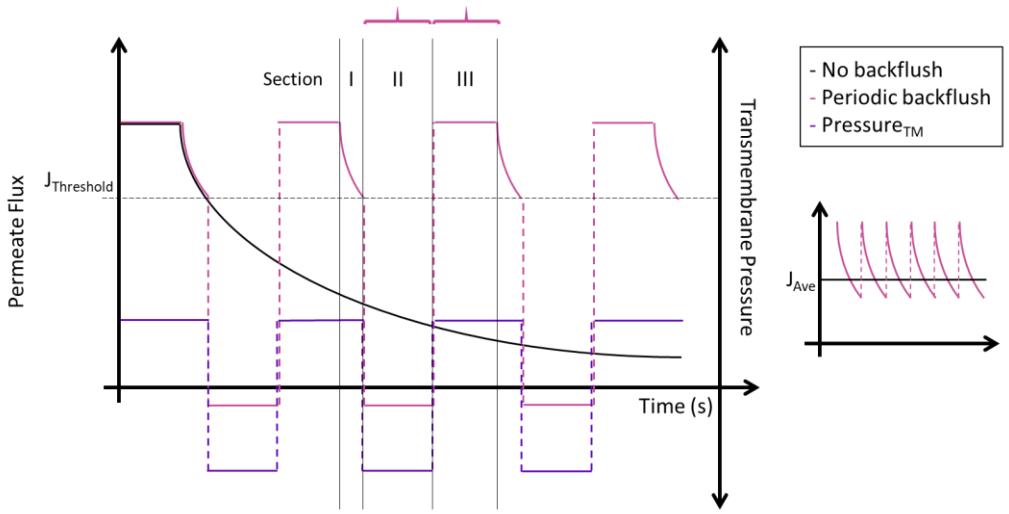


Figure 7 – Effects of PWM control of TMP on permeate flux based on a selected flux threshold. Inducing a negative TMP for a period, Section II, results in a small period of negative flux, ideally restoring the flux capacity of the system. Section I follows exponential decay and fouling as defined for forward flow systems. Section II, as described, creates the anticipated effect in Section III, a region of ideal flux of the permeate back through the filter immediately following the negative flow. In an ideal environment, Section III can be at most as wide as Section II negating both and resulting in an average permeate flux improvement, Subgraph.

Assuming an approximation of the decay in Section I of Figure 7 utilizing Equation (6), and assuming ideal clearance and reintegration of the fouling layer and cake, we can generate a waveform for calculating a theoretical average flux. In order to make the function repeating, we can assume an equation of the general form:

$$p(x) = f(x \bmod N) \tag{9}$$

where p(x) is a periodic function with period of N, which has $p(x) = f(x)\ \forall\ x \in [0, N)$. Under ideal clearance of fouling material with Section II of Figure 7 equal to Section III,

we can calculate the period of repeating function as the difference between α and β resulting in Equation (10).

$$J(t) = J_0 * e^{-f(R_m)*(t \bmod \tau)} + J_{s.s.};\ \text{where}\ \tau = \alpha - \beta \tag{10}$$

The addition of the modulus of t by τ to Equation (6) enables periodic repetition of the equation over a period from [0, τ) for n cycles. Determination of n comes from the duty cycle applied to the volume of solvent processed forward or backward in each cycle resulting in a net forward processing per cycle for a known volume of bulk anticipated to be processed, and can be expressed as:

$$\mathrm{n} = \mathrm{ceiling}\left(\frac{\mathrm{V_T} - \mathrm{V_f}}{\mathrm{V_f} - \mathrm{V_r}}\right) \tag{11}$$

With the ceiling function rounding n up to the nearest integer and V_T being the total expected volume for processing, Equation (11) guarantees a minimum number of cycles with net forward flow $V_f - V_r$ to achieve V_T.

$$V_T = n * (\mathrm{V_f} - \mathrm{V_r}) \tag{12}$$

Therefore, for any $n \in Z^+$ we can calculate the average flux as

$$\mathrm{J}_{ave} = \frac{J_0(1 - e^{-f(R_m)(\alpha-\beta)})}{f(R_m) * (\alpha - \beta)} \tag{13}$$

3.3 Frequency

Frequency is the number of occurrences of a repeating system per unit time. By holding duty cycle constant and modulating either the forward flow volume, backflush volume, α, β, or net volume exchange, we can effectively modulate the frequency of our waveform. In “A new approach to backwash initiation in membrane systems” by Paul Smith et al.[90], the group showed multiple outputs of a feedback-controlled system which varied the processing time and backflush duration of their system holding the proportions of the two flush periods constant, effectively maintaining duty cycle. One conclusion drawn from their study was that smaller forward flush periods produce smaller levels of irreversible fouling. What they failed to comment on, was that the wave they tested with the shortest period produced the worst results. These results imply an optimal frequency for clearance. They were able to show that for large volumes processed, eventually, irreversible fouling dominates all systems and that you can minimize initial flux losses.

Some groups have investigated waveform control or performed specific investigations in narrow ranges in order to minimize backflush volume used in clearance or maximize solvent flux for bulk processes where suspended permeating particles are not the target, but all groups failed to directly control duty cycle in conjunction with flow rate, frequency, and backflush volume combined to maximize permeate flux and filter flux capacity restoration. We can, however, use these investigations to inform investigations in

the future. For example, frequency variations by themselves have been shown to impact recovery and throughput in smaller scale systems.

The effect of pulsatile wave frequency on particle displacement was investigated by Yousang Yoon et al. in "Clogging-free microfluidics for continuous size-based separation of microparticles"[25]. In this paper, the group superimposed a piezoelectric actuated waveform onto a steady forward flow profile to add oscillations to suspended particles and cells to minimize fouling. Yousang observed a 130 Hz ideal oscillation frequency. The group provided no information on reverse flush volumes or compensation for variations in wave frequency but proposed that this frequency may be tied to the resonant frequency of the total fluid mass contained within the microfluidic channel and the connecting tubes. The results of this study imply that frequency should vary with the system design, channel dimensions, the lengths of the inlet and outlet tubes connected with it, and the distance from the channel inlet to the piezoelectric actuation along the inlet tube.

Given the results from this study, we performed initial experiments that show our system requires a time delay compensator prior to adjusting the direction of flow to account for fluidic capacitance in the circuit. The compensator, δ, between forward and reverse flow increases the period of the waveform as described in Equation (14):

$$\lambda = \alpha + \beta + 2 * \delta \tag{14}$$

Chapter 5.4.2 discusses the impact of δ on throughput. We hypothesize that the compensator is proportional to the sum of α and β and can be reduced for smaller forward and backward volumes. This creates effectively higher frequencies and reduces processing time.

3.4 Amplitude

For the purposes of our system, we define the amplitude of our wave as volumetric flow rate. Hong-yu Li et al. through their investigation "Mechanisms by which pulsatile flow affects cross-flow microfiltration" showed the impact that multiple types of waveform have on membrane fouling reduction percentage and relatedly, flux enhancement percentage for cross-flow systems. The group investigated four waveforms and showed that regardless of wave type, the best performance is achieved through rapid changes in fluid velocity but only if the changes result in a negative transmembrane pressure. The best performing waveforms from Li were square wave at 50% duty cycle with dc offset and pulse wave. These two types of waveform can be effectively modelled as square waves with 91 and 94% duty cycle with little to no offset respectively. The investigated waveforms show large rapid changes in TMP can significantly improve clearance, but do not investigate how that changes within different waves or for different filters.

In Chapter 5.4.1, we investigate how the modification of amplitude directly affects duty cycle to improve system throughput. Systems that increases or decreases amplitude of the wave while maintaining reverse and forward volumes directly increase or decrease

the β and α proportions of the wave respectively. For example, doubling the volumetric flow rate of the reverse flow direction will cut the backflush duration (β) in half.

$$\beta = \frac{V_r}{2 * Q(t)} \tag{15}$$

Reductions in backflush direction will also proportionally impact the frequency of the system by scaling the proportions of forward and backward flow: higher flow rates will produce higher frequencies affecting throughput as discussed in Chapter 4.

3.5 Alternate Waveforms

Although not explored experimentally in this thesis, we sought to further our understanding of waveform control outside of square wave systems. In the following sections we present the theoretical basis for controlling both sine and triangle waves. The understanding created herein forms a general framework from which future experiments can be conducted or modelled. Modulation of sinusoids becomes tricky as the functions are continuous. We propose two methods of controlling sinusoids below: periodic piecewise and non-piecewise methodologies.

3.5.1 Periodic Piecewise Sinusoids

A standard sinusoid is originated in the form:

$$Q(\mathrm{t}) = Q_0 * \sin\left(t\frac{2\pi}{\lambda} - c\right) + d \tag{16}$$

Where Q_0 is the peak flow rate of the waveform, z is the period over which the waveform operates and is a positive, real number, c controls the phase shift of the wave, d is the DC offset, and λ is the period of the wave.

The simplest way to modify this waveform is to break it into a piecewise defined function by λ; separating it by the period of the forward flow cycle, λ_1, and the period of the backflush cycle, λ_2. The total period for the combined wave would be the sum of half of each period, and the first period of the wave would be defined as follows:

$$Q(\mathrm{t}) = \begin{Bmatrix} Q_0 * \sin\left(t\frac{2\pi}{\lambda_1}\right) + d; & t \in [0, \frac{\lambda_1}{2}) \\ Q_0 * -\sin\left(t\frac{2\pi}{\lambda_2} - c\right) + d; & t \in [\frac{\lambda_1}{2}, \lambda_3) \end{Bmatrix};\ \lambda_3 = \frac{\lambda_1 + \lambda_2}{2} \tag{17}$$

As a result, we can generate waves of different frequencies dependent on whether Q(t) is greater than or less than zero without knowledge that it is greater than or less than zero. For the purposes of fluidic control, we can set d = 0. Additionally, the first segment has no phase shift, but in order to ensure the wave is continuous, the second piece must have a phase shift of:

$$c = -\frac{\lambda_1 \pi}{\lambda_2} + \pi \tag{18}$$

The phase shift enables the backflush portion of the wave at a higher frequency to start at Q(t) = 0 and represent only the negative portion of the wave from ($\lambda_1/2$) to λ_3 while ensuring continuity of the function. The phase shift is derived by setting the second portion of the piecewise definition above equal to 0 and solving for c at x = ($\lambda_1/2$).

However, piecewise definition makes it difficult to control the wave as a function duty cycle, ϕ, as this would be defined as:

$$\phi = \frac{\int_0^{\lambda_1/2} \left(Q_0 * \sin\left(t\frac{2\pi}{\lambda_1}\right)\right) dt}{\int_0^{\lambda_1/2} \left(Q_0 * \sin\left(t\frac{2\pi}{\lambda_1}\right)\right) dt + \int_{\lambda_1/2}^{\lambda_3} \left(Q_0 * \sin\left(t\frac{2\pi}{\lambda_1} - \frac{\lambda_1 \pi}{\lambda_2}\right)\right) dt} \tag{19}$$

Using Equation (7), we can define the backflush volume as follows:

$$V_r = \int_{\lambda_1/2}^{\lambda_3} \left(Q_0 * \sin\left(t\frac{2\pi}{\lambda_1} - \frac{\lambda_1 \pi}{\lambda_2}\right)\right) dt \tag{20}$$

Solving for the definite integral from ($\lambda_1/2$) to λ_3 with respect to t shows

$$V_r = \frac{-2Q_0}{\pi}\left(\frac{\lambda_1}{2} - \lambda_3\right) \tag{21}$$

From here, we reduce Equation (21) to a single function of λ_2. Defining λ_2 as a function of λ_1 and ϕ gives.

$$\lambda_2 = \lambda_1 \frac{1-\phi}{\phi} \tag{22}$$

And provides the consolidated form of V_r as a function of λ_2

$$V_r = \frac{-2Q_0\lambda_1}{\pi} \tag{23}$$

Combining the above with Equation (9) we can create a periodic piecewise function to control flow in the system.

$$Q(t) = \begin{cases} Q_0 * \sin\left((t \bmod \lambda_3)\frac{2\pi}{\lambda_1}\right); & t \bmod \lambda_3 \in [0, \frac{\lambda_1}{2}) \\ Q_0 * -\sin\left((t \bmod \lambda_3)\frac{2\pi}{\lambda_2} - \frac{\lambda_1\pi}{\lambda_2}\right); & t \bmod \lambda_3 \in [\frac{\lambda_1}{2}, \lambda_3) \end{cases} \tag{24}$$

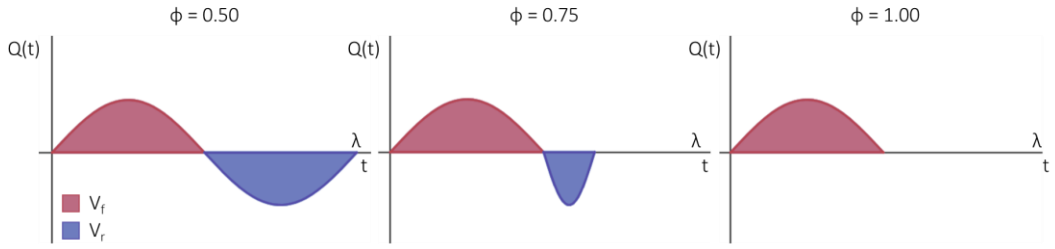


Figure 8 – Periodic piecewise sinusoids for duty cycles of 0.50, 0.75, and 1.00. This sentence serves as a hyperlink to an interactive visualization.

3.5.2 Non-piecewise Sinusoids

If we instead wish to investigate non-piecewise modulation of sinusoid functions, we need to instead vary the amplitude of the backflush portion of the wave described in Equation (16) to achieve variations in duty cycle. In order to maintain the same maximal amplitude, Q_0, of the forward flow region while reducing the backflush amplitude, we need to modify the basal sinusoid by including a DC offset, d, and inversely scaling the entire function by 1+d to maintain the Q_0 resulting in a new function:

$$Q(\mathrm{t}) = Q_0 * \frac{\sin\left(t\frac{2\pi}{\lambda} - c\right) + d}{1 + |d|}; \quad \forall\, d \in [-1, 1] \tag{25}$$

We can then derive the phase shift component, c, as a function of the DC offset to ensure the wave compression and stretch results in a Q(0) = 0 for all DC offset values. Because the DC offset for the base equation is bound to the set [0, 1], we can set the numerator of Equation (25) equal to 0 and solve for c(d).

$$0 = sin\left(t\frac{2\pi}{\lambda} - c(d)\right) + d \;\rightarrow\; c(d) = -sin^{-1}(-d) - t\frac{2\pi}{\lambda} \tag{26}$$

For t = 0 we then have:

$$c(d) = -sin^{-1}(-d) = sin^{-1}(d) \tag{27}$$

Combining c(d) with Equation (25) results in the controlling equation:

$$Q(\mathrm{t}) = Q_0 * \frac{\sin\left(t\frac{2\pi}{\lambda} - sin^{-1}(d)\right) + d}{1 + |d|}; \quad \forall\, d \in [0, 1] \tag{28}$$

Finally, in order to create an overarching controlling equation, we should relate the DC offset to the duty cycle of our system, d(ϕ). The function d(ϕ) is difficult to solve analytically as the α, or time spent in the forward flow portion of the wave, changes as a function of duty cycle. To do this we require a function for the x-intercept for the wave, $t_\alpha(\phi)$, so that we can use Equation (19) and integrate from 0 to $t_\alpha(\phi)$ to determine V_f, and integrate from $t_\alpha(\phi)$ to λ to determine V_r, for all ϕ. Additionally, the Q_0 and $(1+d)^{-1}$ do not vary with time and can be eliminated from the equation, yielding:

$$\phi = \frac{\int_0^{\mathrm{t}_\alpha(\phi)} \left(\sin\left(t\frac{2\pi}{\lambda} - sin^{-1}(d)\right) + d\right) dt}{\int_0^{\mathrm{t}_\alpha(\phi)} \left(\sin\left(t\frac{2\pi}{\lambda} - sin^{-1}(d)\right) + d\right) dt + \int_{\mathrm{t}_\alpha(\phi}^{\lambda} \left(\sin\left(t\frac{2\pi}{\lambda} - sin^{-1}(d)\right) + d\right) dt} \tag{29}$$

With neither $t_\alpha(\phi)$ nor d(ϕ) known, we either need to find one in terms of the other, or replace the equations containing one or the other. We can use a combination of integrals over a full period to determine d(ϕ). Integrating over one period of the controlling equation will give us the difference between the forward flow and backflush portions of the wave.

$$\int_0^{\lambda} Q(t)dt = \int_0^{\lambda} Q_0 * \frac{\sin\left(t\frac{2\pi}{\lambda} - sin^{-1}(d)\right) + d}{1+d} dt = V_f - V_r \tag{30}$$

Additionally, integrating over the absolute value of Q(t) will give us the total volume exchanged through the filter over a single period.

$$\int_0^{\lambda} |Q(t)|dt = \int_0^{\lambda} \left| Q_0 * \frac{\sin\left(t\frac{2\pi}{\lambda} - sin^{-1}(d)\right) + d}{1+d} \right| dt = V_f + V_r \tag{31}$$

Adding Equation (30) and Equation (31) results in $2V_f$. Dividing this sum, then, by two times Equation (31) will also result in a ratio of ϕ without the need to include $t_a(\phi)$.

$$\phi = \frac{\int_0^{\lambda} Q(t)dt + \int_0^{\lambda} |Q(t)|dt}{2\int_0^{\lambda} |Q(t)|dt} = \frac{V_f - V_r + V_f + V_r}{2(V_f + V_r)} = \frac{V_f}{V_f + V_r} \tag{32}$$

Solving for $d(\phi)$ from here is still not trivial as it would involve integrating over an absolute value function and the integral over a full period will change as a function of ϕ to ensure the appropriate proportions of V_f and V_r. In order to solve for $d(\phi)$, a MATLAB script was created to computationally determine the value of d for each ϕ, available in Appendix A.1. The curve fitting tool used in the script produced a best fit curve as described below with an R^2 value of 0.9999

$$\mathrm{d}(\phi) = \frac{2 * \arcsin\big((\phi - 0.5) * 2\big)}{\pi} \tag{33}$$

For a better understanding of Equation (33) we can investigate the shifters and scalars. The function d(ϕ) spans values from -1 to 1 with duty cycle ranging from 0 to 1. Therefore, the values of ϕ must be shifted down by 0.5 and scaled by 2. After taking the inverse sine of the shifted and scaled axis values, we can then scale the values of d(ϕ) to a range of -1 to 1 by multiplying be 2 over π. With a function for d(ϕ), we now can now generate waveforms for various duty cycles ranging from 0.00 to 1.00.

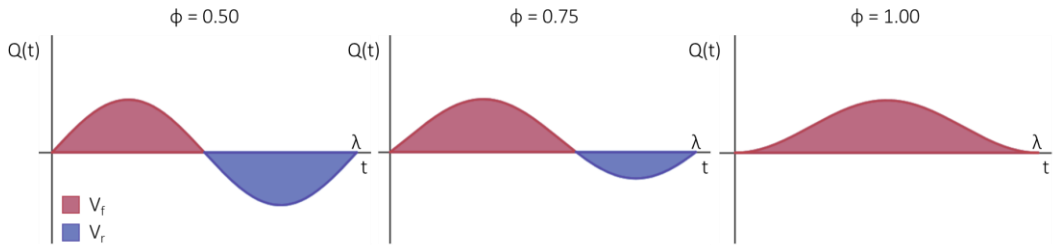


Figure 9 – Non-piecewise sinusoids for duty cycles of 0.50, 0.75, and 1.00 achieved through stretching and compression. This sentence serves as a hyperlink to an interactive visualization.

3.5.3 Periodic Triangle Wave

Rearranging Equation (7), we derive the following equation which establishes a relationship between the forward flow volume and backflush volume.

$$V_f = \frac{\phi}{1-\phi} V_r \tag{34}$$

For the purposes of creating a triangle wave under our constraints, V_f and V_r can be represented as triangles of areas:

$$V_f = \frac{b_f h_f}{2};\ V_r = \frac{b_r h_r}{2} \tag{35}$$

Where b_f and h_f, equal to α and Q_0, are the base and height, respectively, of triangle V_f, and b_r, equal to β, and h_r, are the base and height, respectively, of triangle V_r. In order to obtain a triangle wave where the magnitude of the slope of the line is identical throughout the wave, we can define the slope as follows, where m is the magnitude of the slope of the line.

$$m = \frac{h_f}{b_f/2} = \left|\frac{h_r}{b_r/2}\right| \tag{36}$$

We can then divide each period of the wave into three sections separated by the peak and the trough. These three periods are from zero to half of α, half of α to α plus half of β, and from α plus half of β to λ where λ is α plus β. We can then generate a triangle wave as a function of the slope over each of these sections as follows:

$$Q(t) = \begin{cases} \frac{2h_f}{\alpha}(t \bmod \lambda); & (t \bmod \lambda) \in [0, \frac{\alpha}{2}) \\ -\frac{2h_f}{\alpha}\left((t \bmod \lambda) - \frac{\alpha}{2}\right) + h_f; & (t \bmod \lambda) \in [\frac{\alpha}{2}, \alpha + \frac{\beta}{2}) \\ -\frac{2h_f}{\alpha}\left((t \bmod \lambda) - \left(\alpha + \frac{\beta}{2}\right)\right) - h_2; (t \bmod \lambda) \in [\alpha + \frac{\beta}{2}, \lambda) \end{cases} \quad (37)$$

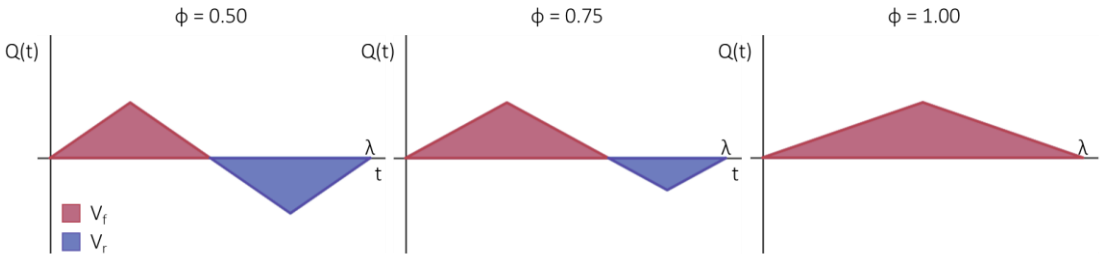


Figure 10 – Piecewise defined triangle wave for duty cycles of 0.50, 0.75. and 1.00. This sentence serves as a hyperlink to an interactive visualization.

3.6 Discussion

To conduct PWM, PAM, and PFM experiments, we defined variables for the three components that control the actuation of square waves in fluidic circuits: duty cycle, periodicity, and amplitude. We explore the literature and address the accomplishments and shortcomings of the leaders in this space. We base the selection of square and pulse wave control for our system on the observed cake resistance reduction percentage in cross flow systems associated with rapid oscillatory direction change and negative TMP creation. The use of square wave flow rates should limit the factors effecting clearance to backflush volume as opposed to a combination of volume and rate of change of flow rate. We proposed to increase permeate flux, throughput, and particularly recovery percentage in

dead-end systems by implementing a PWM periodic backflush fluid flow cycle to clear reversible fouling, blocking, and cake through the control of duty cycle, flow rate, backflush volume, and frequency.

Additionally, we lay the theoretical foundation and general framework for control and execution of two types of sinusoidal waves and triangle waves for future experimentation and modelling. We hypothesize that these waveforms will likely underperform with lower rates of change of flow rate. We can conclude, however, that piecewise periodic sinusoidal control would likely outperform the other waves because backflush through this control algorithm will approach a system similar to a pulse as higher duty cycles, however, this will likely underperform square wave systems as these pulses only occur for small backflush volumes losing the benefits of both.

CHAPTER 4. PULSE WIDTH MODULATION

4.1 Abstract

Standard dead-end sample filtration is used to improve sample purity, but is limited as particle build-up fouls the filter, leading to reduced recovery. We show the fouling layer can be periodically cleared with backflush algorithms applied through a customized fluidic actuator using variable duty cycles, significantly improving filtration efficiency. A Pulse Width Modulation (PWM*)* process was used to control the flow profile and periodically backflush the filter membrane to repeatedly interrupt cake formation and reintegrate the fouling layer into the sample, improving net permeate flux per unit volume of sample by restoring filter flux capacity. To evaluate the flow control, a PWM flow was applied to suspensions containing 2.19 um and 7.32 um polystyrene microbeads and filtration produced 18-fold higher permeate concentration, higher recovery up to 68.5%, and an 8-fold enrichment increase, compared to a uniform flow profile. As the duty cycle of flow control approaches 50%, the recovery percentage monotonically increases after a critical volumetric threshold defined by duty cycle. Improved recovery was shown to decrease throughput, Equation (4), by increasing processing time asymptotically compared to linear growth of recovery percentage. To address the throughput tradeoffs, we developed and validated a mathematical model to evaluate how fast, small-volume backflush pulses using a near 50% duty cycle can yield higher recovery by decreasing obstruction and reducing the time-average flow resistance associated with a fouling layer and maximizing forward

volumetric flow velocity. An optimized PWM flow control was then used to purify custom particles for immune activation, achieving 3-fold higher recovery percentage and providing a new route to improve purification yields for microparticle recovery and cell and tissue therapies, CHAPTER 6.

4.2 Introduction

Dead-end filtration using patterned microsieves, fiber meshwork, and membranes of various materials is a standard technique to isolate desired particles of various sizes and is often used in clinical and laboratory settings for therapeutic and diagnostic applications[26–29]. A common practice is to use microparticles to bind and filter out antibody excretions from laboratory and human derived cells[53,54,59,92]. Both biological and physical suspensions can be filtered to yield high purity and enrichment at a high throughput. Dead-end filters are especially susceptible to fouling, however, which leads to lower recovery percentage and yield as a direct result[9,13,25,26]. Because a decreasing yield negatively impacts therapeutic quality[64,93,94], clinical and industrial therapeutic manufacturing will frequently change or increase the surface area of the dead-end filter[90], or switch to crossflow filtration modalities[1,22], which further decreases the throughput and increases processing time.

Membrane fouling is caused by pore blocking followed by cake layer formation, resulting in an exponential decay with time in the flux of particulate[40,77,78,80,95–97]. Membrane fouling also affects crossflow systems and, in this case, numerous studies were conducted to disrupt cake formation and reintegrate particulates into the bulk flow feed

stream[1,2,22,35,73,98,99]. For example, crossflow filtration can disrupt caking by implementing an oscillatory flow with a sinusoidal flow velocity or a pulsatile flow, consisting of a steady flow with oscillations superimposed across the membrane potentially creating negative transmembrane pressures[2]. These studies examined the effects of numerous waveforms, including variations of sinusoids, pulses, and square waves[1,2,99].

Oscillatory and pulsatile techniques showed improved clearance of the crossflow membranes, leading to an increase by up to an order of magnitude of flux of permeate across the membrane. There is now a consensus in crossflow systems that reversals in transmembrane pressure (TMP) for short periods of time disrupt accumulated particulates and reduce membrane resistance, improving permeate flux for the processed samples[1,4,99]. However, crossflow filtration suffers from extraordinarily low throughput and requires recirculation of processed samples to achieve higher recovery percentages, straining cellular samples and impacting their morphology and function.

Here, we investigate controlled reversal of TMP through PWM volumetric flow rate to analyze the effects of periodic backflush on dead-end filtration systems. The lack of research in here can be attributed to a couple reasons. Firstly, dead-end systems lack shear flow removal of cake fouling which implies limited benefit in low frequency high forward volume proportion systems, the only investigated to date. Secondly, the time contribution of PWM approaches on total processing time grows asymptotically compare to linear growth of recovery percentage, potentially making throughputs worse than cross-flow

standards. Because dead-end filters are ubiquitous in clinical and laboratory settings, solving the fouling problem for this type of filter will be impactful, and doing so using PWM is significant and novel as it pursues investigational opportunities not previously investigated.

We explore how periodic backflush can be applied to dead-end filtration systems and optimized through manipulation of square wave fluid flow control to improve recovery percentage, while maintaining enrichment, purity, and throughput. A schematic of the procedure is shown in Figure 11. The flow is controlled through pulse width modulation (PWM) control of a syringe pump to manipulate the fluid velocity. We characterize the PWM control by periodicity and duty cycle, Figure 2. We show the use of PWM backflush for two experimental modalities, Fixed Backflush (FBF) and Fixed Forward Flush (FFF), compared to a uniform forward flow. These experimental modalities allow us to explore the effects of cake thickness on clearance and backflush volume on reintegration by holding the backflush volume constant and varying the forward flow regime, affecting cake buildup, and by holding the forward flow regime constant while varying backflush, affecting reintegration, respectively. PWM applied to dead-end filtration produced over 18-fold higher permeate concentration, significantly higher recovery percentage of up to 68.5% and increased enrichment of 8-fold. Further, we built a computational model including the interruption of cake formation and reintegration of the fouling layer into the bulk of a sample during PWM periodic backflush, resulting in an improved net permeate flux per unit volume of sample while restoring flux capacity to the filter.

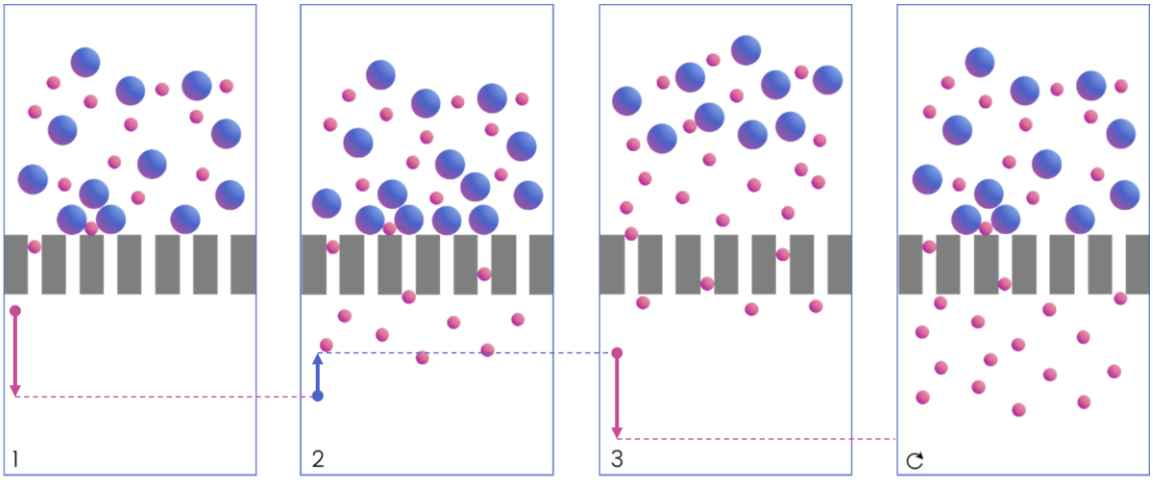


Figure 11 – Microfiltration model with periodic backflush. Arrows indicate fluid velocity direction and duration in each step leading the result depicted in the subsequent step. Step 1) forward flow resulting in cake formation and fouling as pore blocking occurs. Step 2) initiation of backflush to remove particle buildup from the cake layer and membrane surface. Step 3) re-initiation of forward flow after backflush has cleared the fouling. Negative flux of permeate particulate occurs during step two and reaches a maximum at start of stage 3. This is repeated periodically in this order to improve flux and increase permeation and recovery percentage of processed samples.

4.3 Methods

4.3.1 Membranes

The experiments were conducted with Pall Acrodisc syringe filters with Versapor (hydrophilic polypropylene). The filters are commonly used for both aqueous and organic samples with a glass-fiber prefiltration component. All filters have a diameter of 25 mm

and functional cross-sectional area of approximately 2.8 cm^2 and are loaded into the system as shown in Figure 12 A.4.

4.3.2 Flow Control System

4.3.2.1 Mechanical System Design

A custom syringe actuator, Figure 12, was constructed to inject preprogrammed PWM pressure waveforms. The filtration system, composed of a linear actuator, syringe holster, TMP measurement sensor, and magnetic mixer, is mounted vertically. The feed reservoir is open to atmospheric pressure, and the linear actuator manipulates flow by alternating the syringe pump, resulting in pressure driven flow movement direction from the permeate side outlet reservoir, Figure 12.

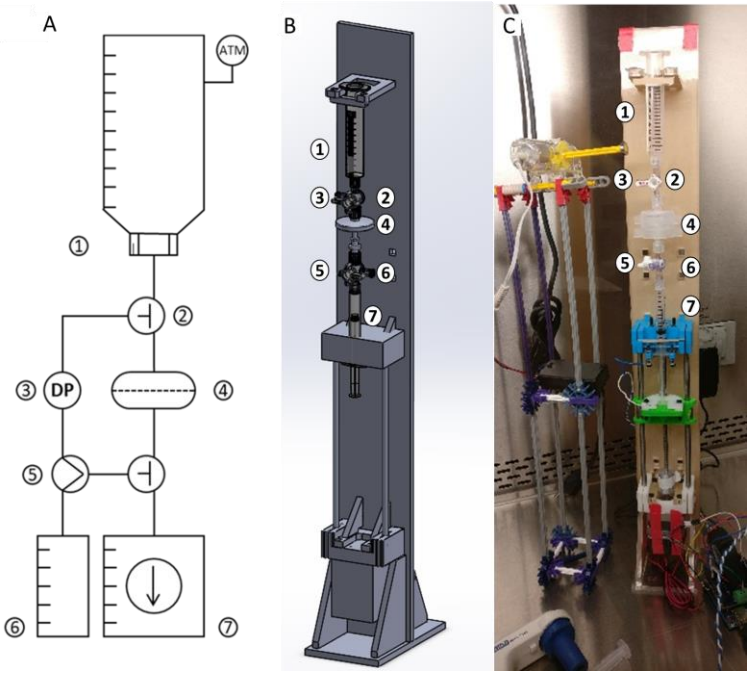


Figure 12 – Provides a A) Process Flow Diagram, B) 3D rendering, and B) executed PWM programmable syringe actuator prototype, including: 1. sample reservoir, 2. four-way stopcock valve, 3. differential pressure sensor, 4. dead-end filter, 5. three-way valve, 6. priming fluid reservoir, and a 7. pressure generating permeate reservoir.

The actuator executes syringe displacements in discrete increments with a minimum resolution of 0.23 µL when using a 3-mL BD syringe. The software records inputs of actuator speed, duty cycle percentage, gross volume exchange, and total volume of sample to process. All experiments were conducted at a positive and negative absolute flow rate of 4.55 mL/min, which was found to minimize pressure fluctuations determined through testing with a differential pressure sensor. The system utilizes a reservoir for holding up to 10 mL of sample open to atmospheric pressure, can integrate any Luer-lock based

commercially available syringe filters, and measures transmembrane pressure at a sample rate of 50 Hz.

4.3.2.2 Software

A custom user interface, designed in LabVIEW, communicates with an Arduino Uno R3 with custom motor shield to control a stepper motor which in turn translates rotational motion to linear actuation of a syringe, Appendix A.3 and A.4. The fluid velocity, pulse duration, duty cycle (ϕ), and frequency of the flow cycle are controllable through the LabVIEW interface. In this system, because flow rate will be held constant for all experiments, periodicity is representative of the gross volume exchange across the filter within a given cycle. Duty cycle, on the other hand, is representative of the ratio of forward flow, V_f, to backflush, V_r, and is proportional to net volumetric flux per cycle.

Process parameters are explored by holding the volume exchanged in either the backflush or forward flow components of the duty cycle constant and varying ϕ. By holding the backflush volume of the duty cycle constant, variations in ϕ result in a changing periodicity that enables us to draw conclusions about the effects of cake layer buildup on reintegration for a given backflush volume. Also, by holding forward flow volume constant and varying ϕ, we can draw conclusions on how backflush volume affects cake disturbance and reintegration for a given cake deposition rate.

4.3.2.3 Circuit Design

The system uses three main components to manipulate the actuators: an ATmega 328P-PU Arduino Uno microcontroller; a stepper motor-based pump actuator that converts rotational motion into linear actuation; and a custom motor shield. The motor shield utilizes a stepper motor driver and integrates a barrel connector to receive +12V 2000 mA power, a transimpedance amplifier circuit, and pull-down resistors for communication and sensing. The circuit diagram available in Appendix B.

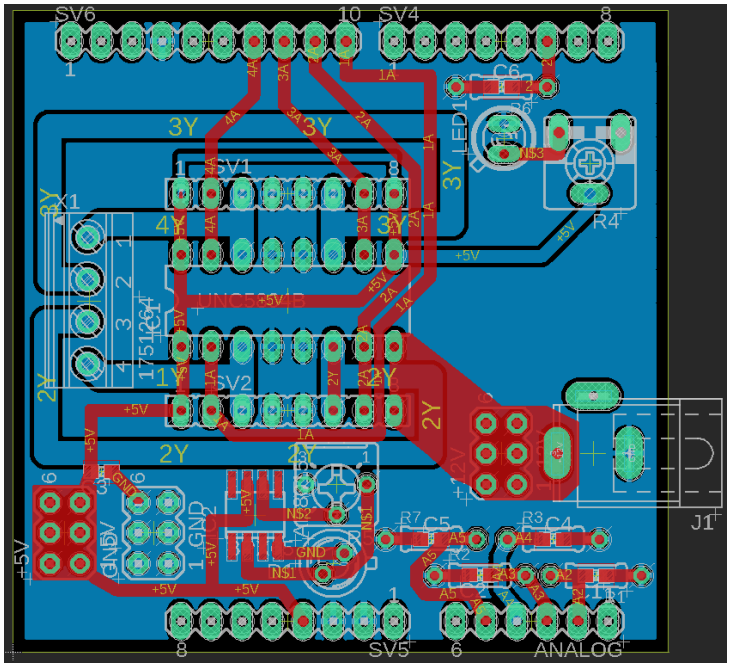


Figure 13 – Diagram showing custom motor shield for Arduino based control of fluid actuator circuit

4.3.3 Waveforms

The system contains an integrated differential pressure, MPXV7025DP-ND, sensor to directly measure changes in TMP ±3.63 PSI, Figure 12. This measurement directly

correlates to changes in total membrane resistance (R) as a function of cake layer deposition (R_f).

$$TMP = J_T * \left(R_m + R_f\right) * \mu_T \quad (38)$$

Where:

J_T = total solute flux across the membrane

μ_T = fluid viscosity

For both fixed backflush and fixed forward flow volume experiments, square wave actuation of a 3-mL syringe was used to create a square fluid velocity profile measured indirectly through TMP with negative backflush pressures compared to baseline. Duty cycle was varied from 0.55 to 1.00 in increments of 0.05. A one second compensator is added between each change in volumetric fluid flow direction. An example of TMP wave profiles is provided in Figure 14 for a DI sample at duty cycles of 1.00 and 0.75. This characterization waveform is descriptive of fluid behavior in the system.

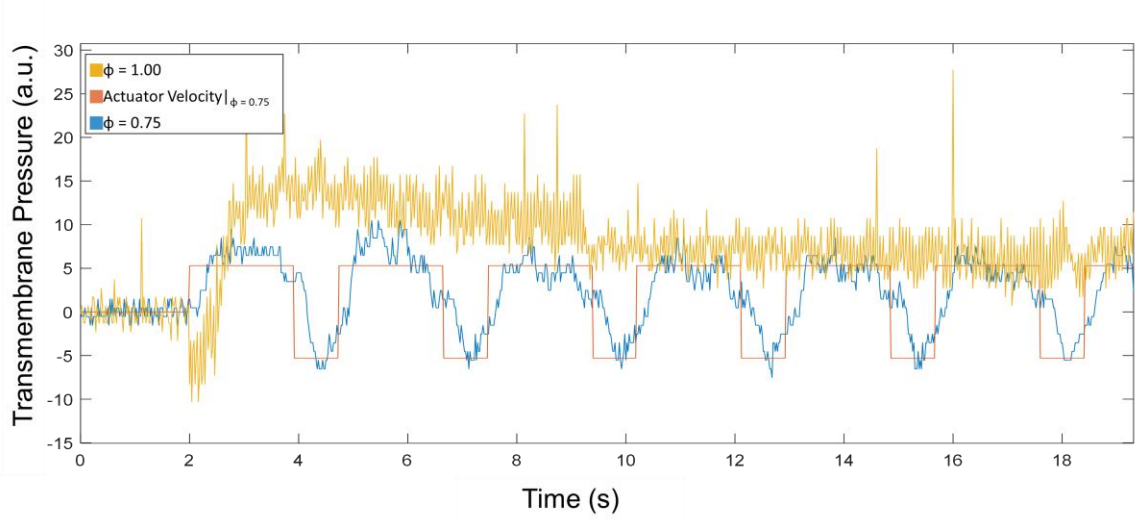


Figure 14 – Pressure waveform profile showing transmembrane pressure for duty cycles ϕ = 1.0 and 0.75 for square wave fluid velocity input.

4.3.4 Particulate sample processing

Mixtures of polystyrene (PS) microspheres were suspended in phosphate buffered saline (PBS) solution with 0.02% tween by volume and were transferred to the sample reservoir for processing by the filtration system. The particles included mixtures of 2.19 µm and 7.32 µm average diameter PS particles (Bangs Labs) in a 50:50 ratio at a concentration of five million total particles/mL, 2.5 million particles/mL each, Figure 15. Particles were flowed through the filter using a syringe pump. Fixed backflush and fixed forward flush volume experiments used to process 1 mL of sample fluid at an absolute flow rate of 4.55 mL/min. For each experiment, the testing rig is primed with DI water prior to testing to eliminate bubbles and reduce capacitive actuation effects created from the expansion and compression of air in the chamber.

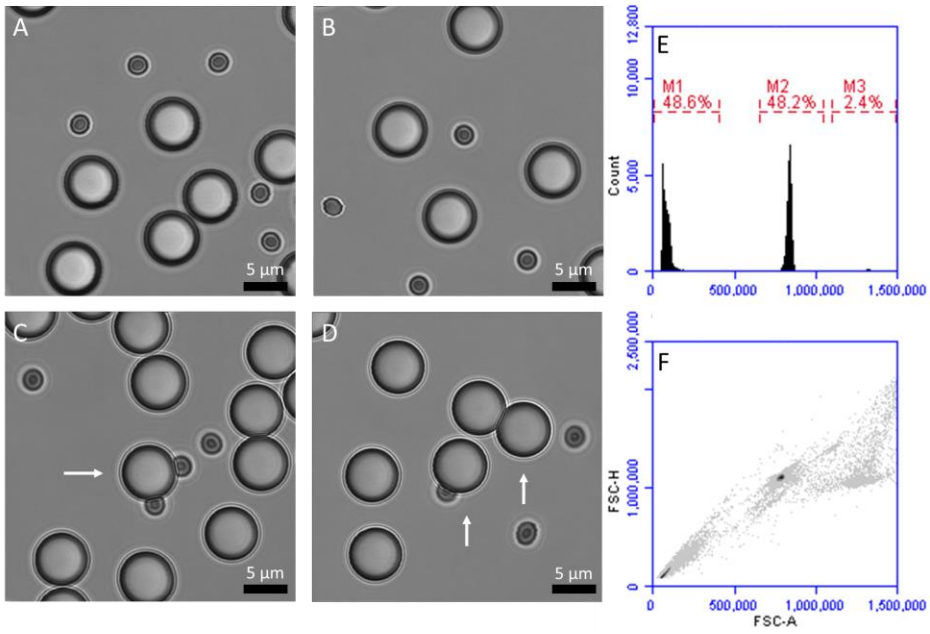


Figure 15 – Microsphere suspension confocal images (scale bar 5 µm) and population distribution flow cytometric results. A) Shows standard mix of microspheres. B) Variance in concentration can result in slightly more dilute or concentrated suspensions. C) and D) Less than three percent of the population consists of particles bound to other particles (indicated by white arrows). This population can be seen in the forward scatter histogram E and forward scatter height vs area plot F.

4.3.5 Volumetric Normalization

After processing the total volume of microsphere suspension samples through the membrane using a PWM technique, all samples were normalized for volumetric differences in the priming stage arising from minor variations in the negative space of the stopcocks and filters using the following procedure. The output from both permeate and retentate were collected in 5 mL culture tubes and centrifuged at 3500g for 15 minutes. The supernatant was removed, and the particle pellets were resuspended in 3.4 mL of DI

water. All samples were then vigorously vortexed and kept on a tube rotator to break up the pellet and ensure uniform dispersion through the solution. The particle count for each condition was then measured using flow cytometry.

4.3.6 Microfluidic Devices

To investigate membrane dynamics and the effects of backflush on caking material, we created microfluidic channels via PDMS molding to visualize backflush with high-speed and fluorescent cameras. The devices are single channel rectangular prism or cylindrical molds created by suspending either a square cross-sectional wire or 22-gauge circular cross-sectional wire between laser cut mounts in a petri dish. PDMS base and curing agent are mixed and poured over the wires. The PDMS was cured at 60 °C for one hour. After curing, the devices were cut from the mold and wires removed after soaking in ethanol.

Inspired by the work performed by Zeng et al.[100] we cut the channels down the middle about 90% of the way through the mold to insert a polycarbonate membrane[101]. To bond polycarbonate to PDMS, surface modification had to be done on both substrates to create silanol groups, which will irreversibly bond to each other on contact[102]. After oxygen plasma treatment, the polycarbonate membranes were submerged in a 5% (3-Aminopropyl) triethoxysilane (APTES) solution at 80°C. The functionalized polycarbonate surface molecules react with APTES transferring the silicon-carbon-amine chain of the APTES molecule to the polycarbonate surface. When treated with oxygen

plasma a second time, polycarbonate can bond with functionalized PDMS to form siloxane groups, permanently bonding the two materials together. Polycarbonate membranes were boiled in a 5% APTES solution while the PDMS chips, taped open, were exposed to oxygen plasma, ensuring straight from plasma generator bonding.

4.4 Results

4.4.1 Fixed Backflush

FBF volume flow was conducted to hold backflush constant and independently examine the effect of forward flow durations. From Figure 16, we know that longer forward flow periods result in larger volumes of bulk solution moved through the membrane for each period. Longer forward flow periods result in increased caking and decrease the ability to clear the fouling due to packing, adsorption, and higher membrane resistance. We hypothesized that increased cake thickness should decrease fouling clearance and reintegration, resulting in decreased average permeate flux for a given net volume of processed bulk solution. In order to measure this, we designed fixed backflush volume experiments, described in Figure 16A, in which we modify the waveform duty cycle while maintaining backflush volume constant.

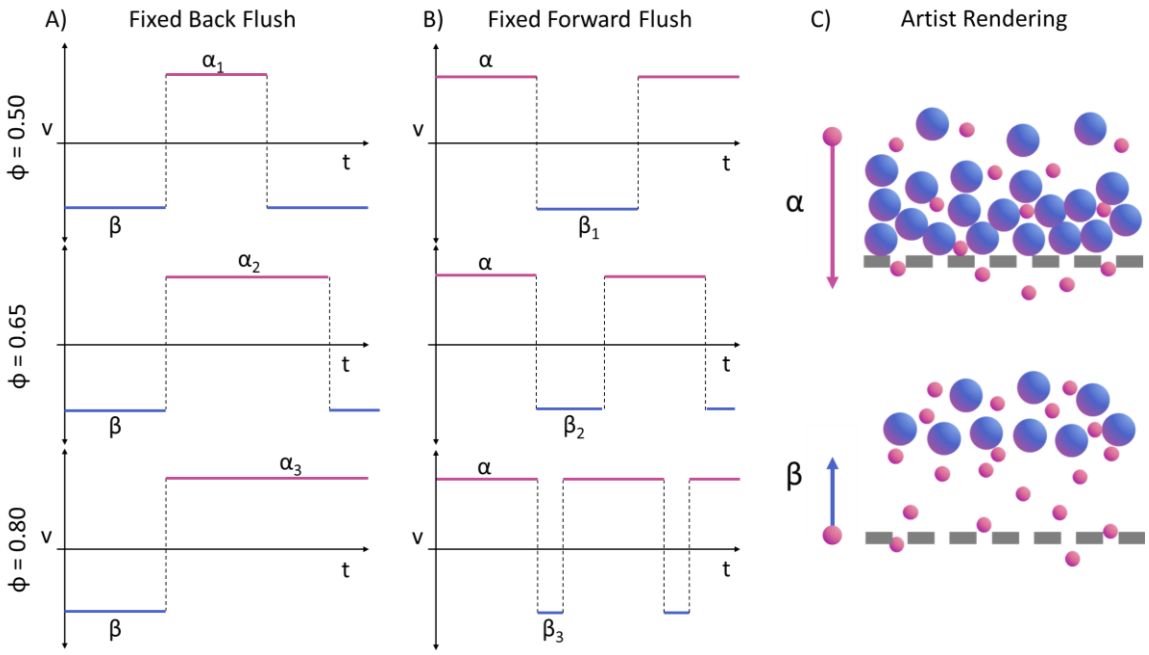


Figure 16 – A) Representation of how variations in duty cycle percentage correspond to changes in forward flow duration for fixed back flush. This enables us to hold backflush volume (β), observable in artist rendering, constant while investigating the impact of cake thickness, proportional to forward flow volume (α). B) Representation of how variations in duty cycle percentage correspond to changes in backflush duration. This enables us to hold cake thickness to a consistent value while investigating impact backflush volume. C) Artist rendering of forward and back flush on fouling layer.

The duty cycle of the backflush process was found to impact the target particle concentration at the permeate side of the filter. Figure 17 shows the enrichment of smaller 2.19 µm particles from larger 7.32 µm particles initially mixed together and processed by the filtration system described in Figure 12, for different duty cycles (ϕ). ϕ for this analysis is defined as forward flow volume divided by the gross volume exchange through the filter or the sum of forward flow plus back flow volumes, Equation (48). In the retentate side

row of Figure 3, the targeted 2.19 µm particle population is shown to decrease as ϕ approaches 0.55. In the permeate side, it is observed that the 2.19 µm particle counts increase significantly without noticeable increase in 7.32 µm particles as ϕ approaches 0.55.

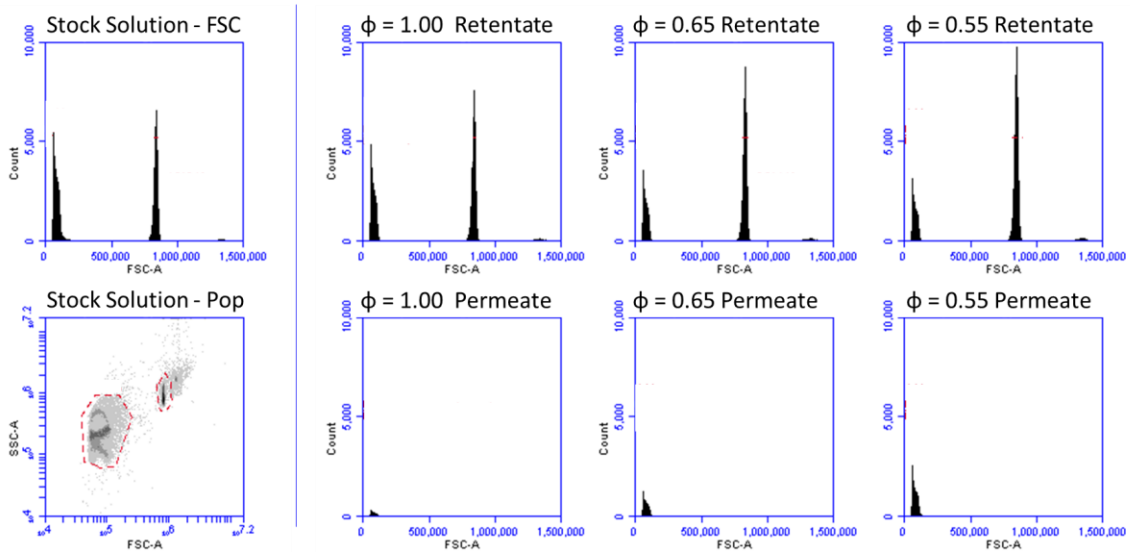


Figure 17 – Flow cytometry data showing forward scatter histograms and population dynamics. Images left of dividing line correspond to stock solutions with 2 µm and 7 µm polystyrene microspheres in a 50:50 ratio. Right of dividing line, rows show retentate side and permeate side population dynamics for duty cycles of 1.0, 0.65, and 0.55 respectively in each column. It can be observed that as duty cycle decreases, the 2 µm population quantity decreases in the retentate and increases in the permeate.

Particle counts from individual experiments resulting in the images in Figure 17 were used to calculate a concentration fold change curve. The mean particle concentration between ϕ = 1.0 and 0.95 was calculated, and the concentration data for all ϕ were normalized to this mean to express fold change improvement as a function of ϕ, shown in Figure 18. We find an average fold change improvement of 7.82 for ϕ = 0.55.

Recovery percentage, Figure 18B, shows the number of desired or targeted particles collected at the output compared to the number of desired particles initially supplied at the input, Equation (3). The recovery percentage significantly increased (pairwise comparison with t-test $p < 0.0001$) from a median of 7.11% to 54.73%, for $\phi = 1.0$ and 0.55 respectively. Figure 18B shows that recovery percentage follows a bi-linear function with a region of little to no gain from $\phi = 1.0$ to 0.8 followed by a region of linear gain from $\phi = 0.8$ to 0.55.

Further, microsphere counts were compared to original sample stock to calculate recovery percentage and population percentage changes were used to calculate the enrichment factor[17] as functions of ϕ. Enrichment factor is a ratio of ratios that shows the proportion of targeted to nontargeted particles in the output compared to the initial sample being processed at the input, Equation (2). We used enrichment factor as a metric to show that this process does not significantly increase undesired particle counts at the output in conjunction with desired particles.

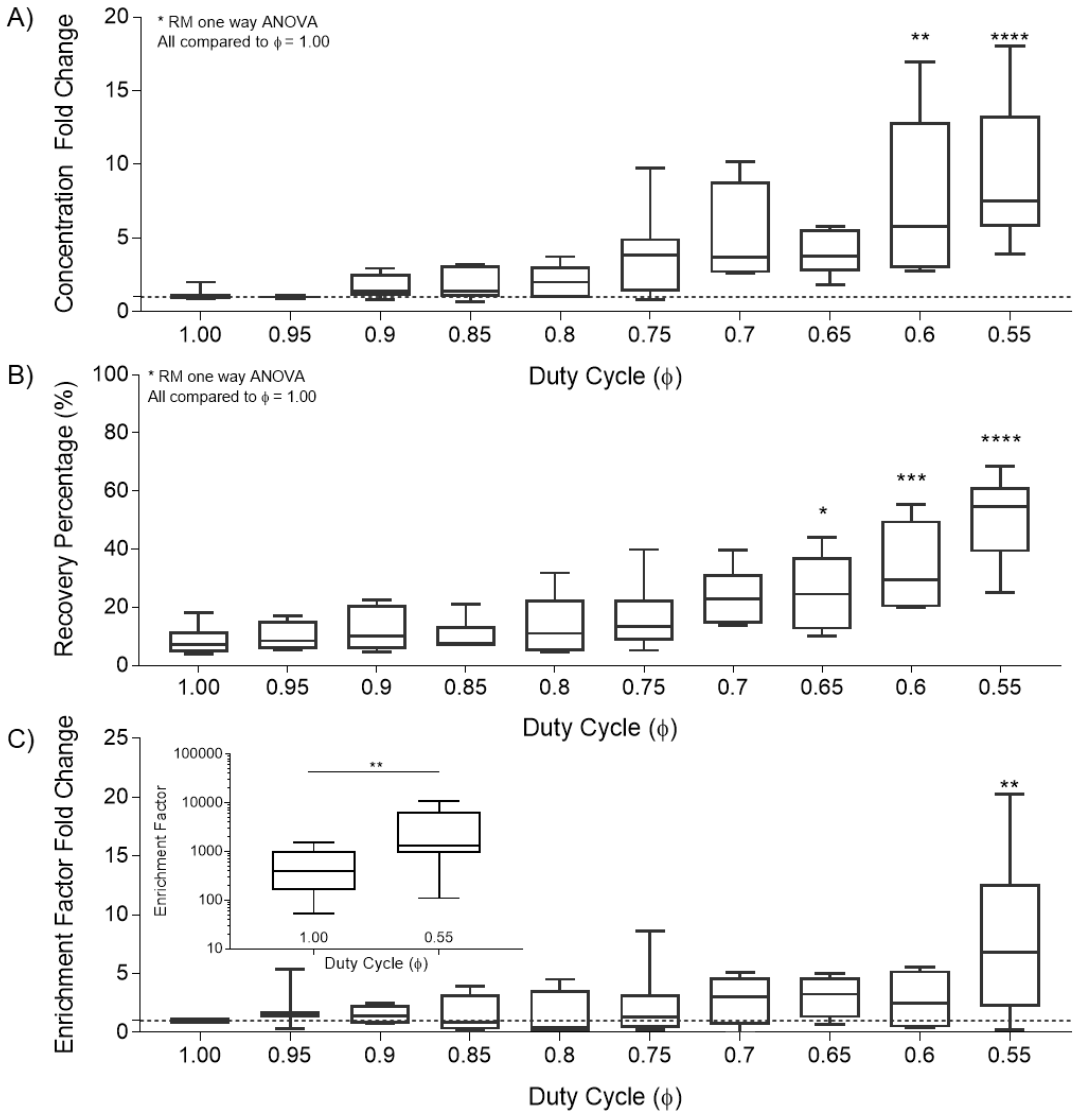


Figure 18 – A) Fixed Back Flush concentration fold change as a function of ϕ. Data for n = 10 experiments each were normalized to the concentration at ϕ = 1.00. Dotted line shows a concentration fold change of 1, corresponding to conventional ϕ = 1.0. B) Fixed Forward flow Recovery percentage as a function of duty cycle. C) Enrichment Factor Fold Change with respect to duty cycle fraction 1.00. The inset shows the numerical result of Enrichment Factor between ϕ = 1.00 and 0.55. Significance data calculated for multiple comparison test against the indicated mean. P-values decrease with increasing asterisk count indicating less than 0.05, 0.01, 0.001 and 0.0001, respectively.

Increasing values of enrichment indicate increasing effectiveness of processing. We observe that increased enrichment factor results from increased recovery percentage of targeted particles at the output with no significant change in the nontargeted output particle counts. Figure 18C shows enrichment significantly improved ($p < 0.01$) as ϕ approached 0.55, increasing from an average of 567.3 to 3374.8 for $\phi = 1.0$ and 0.55, respectively, Figure 18C.

4.4.2 Fixed Forward flow

The total backflush volume, affecting particle displacement from the filter, can impact breakup and reintegration of the fouling layer. To control for the impact of backflush volume on reintegration of fouling layer particulate, we fixed the forward flow volume while varying the backflush volume as a function of ϕ, illustrated in Figure 16. The total forward flow volume was fixed to a predetermined level to hold cake material buildup consistent in these experiments. The FFF volume was chosen from the FBF volume values which produced minimal but observable changes in concentration. The output concentration increases as the backflush volume increases and these data were normalized as in Figure 18 to calculate concentration gain, shown in Figure 19. However, concentration gain saturates at a duty cycle fraction of 0.75 and recovery percentage does not significantly improve beyond this duty cycle value (Figure 19). Similar to our observations in the FBF results, we observe that smaller periodicities result in improved clearance, but with diminishing net particle flux with larger backflush volumes. We

conclude that there exists an optimal recovery percentage that results from the increasing periodicities explored in FBF and the decreasing periodicities of FFF at a duty cycle of 0.55 for the tested backflush volume. Based on this conclusion, we developed a mathematical model to parameterize the results and find optimal enrichment conditions.

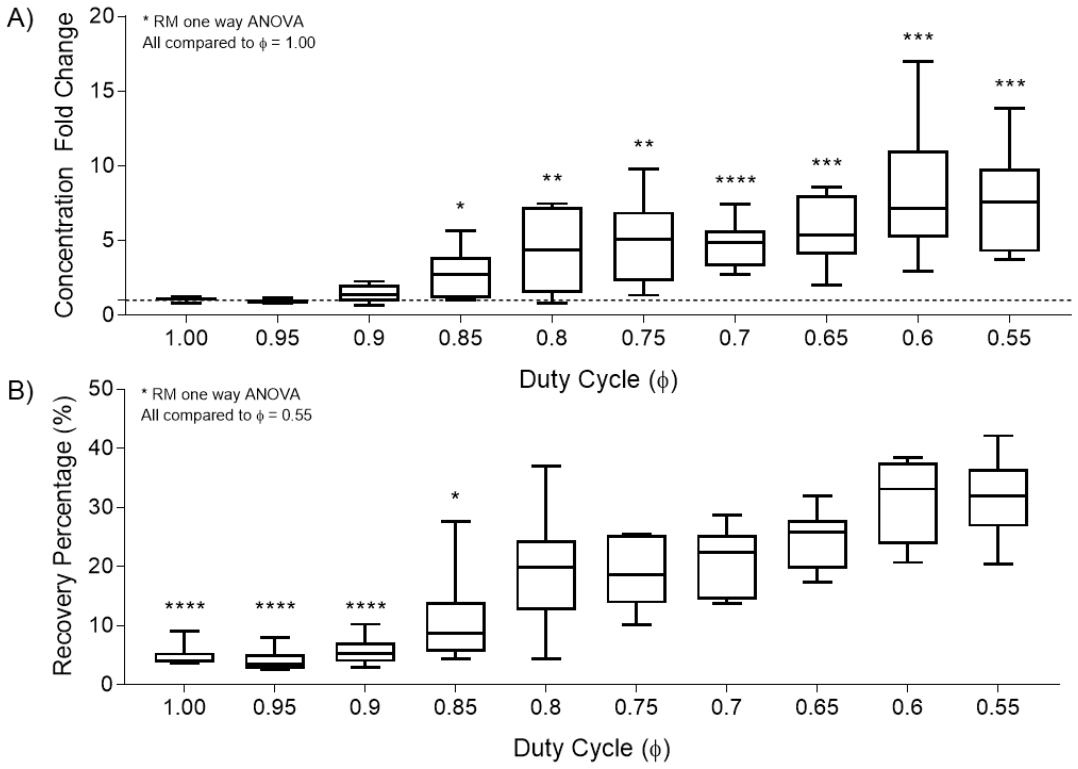


Figure 19 – A) Fixed Forward flow concentration gain as a function of duty cycle percentage from 100 to 55 for n = 10 fixed forward flow volume experiments. B) Fixed Forward flow Recovery percentage as a function of duty cycle.

4.4.3 Particle Dynamics

To better understand the processes occurring during backflush, we record video via high-speed and fluorescent microscopes. The high-speed camera enabled us to capture the effects of different duty cycles on cake layer formation for PWM FBF experiments. As shown in Figure 20, significant cake layer formation can be observed in the $\phi = 1.00$ case with nearly imperceptible cake formation in the $\phi = 0.55$ case at similar time points in processing. In these experiments, it is difficult to track individual microspheres from the bulk or cake layers and we can only make judgements on bulk processes. We conclude from these video sets that frequent and large proportional volumes of backflush help to reduce or eliminate caking by providing opportunity for particles that would have normally been rejected by the cake layer resistance to no longer be rejected. From this we can assume that periodic backflush not only clears the fouling layer, but also reintegrates, to some extent, the cake layer into the bulk flow.

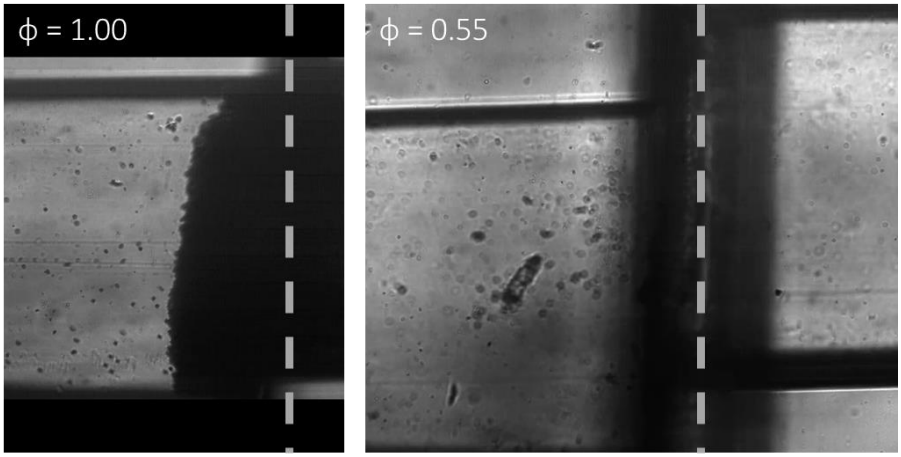


Figure 20 – High speed camera capture of fouling layer and cake for $\phi = 1.00$ and $\phi = 0.55$ at similar processed volumes. A membrane location indicator was placed on images. Significant buildup is visible on $\phi = 1.00$ with negligible buildup on $\phi = 0.55$.

To investigate reintegration of caking into the bulk with periodic backflush, we partnered with Cameron Yamanishi of the Shuichi Takayama Lab at Georgia Tech to apply a particle tracking software to study the impact of backflush on membrane fouling. To record fluorescent particle z-position using the tracking microscope, we scaled down the internal side length of the filter area to 250 μm and used a square cross-section. We capture fluorescent video of particle movement for multiple periods, shown in Figure 21.

We observed that each backflush cycle moves different particles and moves the new and previously moved particles to new locations in the channel. We attribute the changes in particles removed and particle displacement to variation in surface pressure differentials and detachment events. This change in pressure around the membrane and the associated variation in detachment time and location, can be attributed to convective currents and redistribution of large particles on the membrane at the end of each backflush event.

Figure 22 shows the fluorescent microscopy images and corresponding z-position, or height within the channel, with time for consecutive backflush events. A clear difference in z-axis distribution can be observed during backflush events compared to forward flow over time. During backflush, particles are shown to express virtually no displacement along the x-axis, true for systems operating in laminar flow regimes. However, we show vertical particle displacements of up to 15 μm, greater than two particle diameter lengths, suggesting convective currents are present in the y-z plane. We claim the observed backflush displacements are associated with convective currents and not Brownian or

chaotic flow regimes because these migratory behaviours are not present during forward flow phases. Forward flow is shown to have little to no deviation during particle transit toward the membrane. The difference in displacements between backflush and forward flow results in a redistribution of particles on the surface, and redistribution, in turn, likely enables the previously rejected target particles the ability to permeate the membrane.

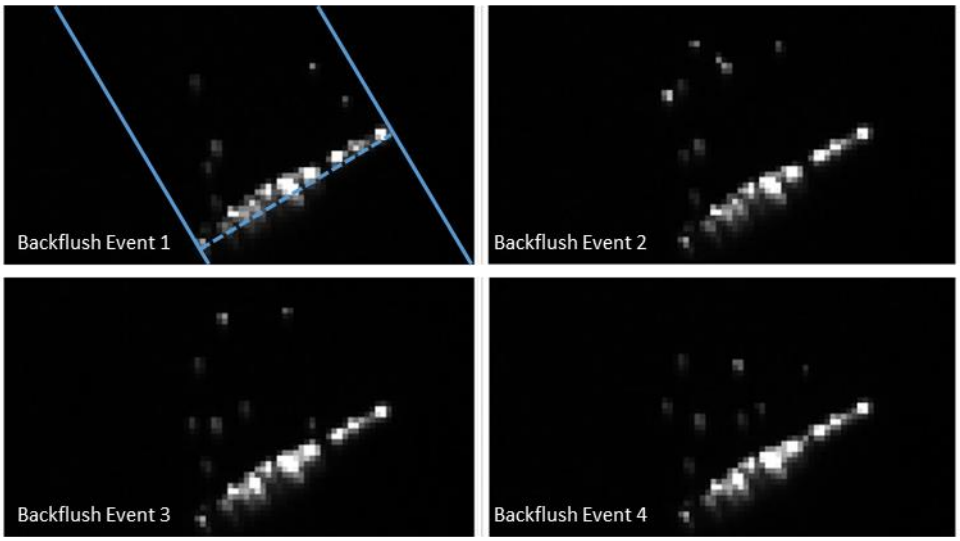


Figure 21 – Fluorescent microscopy images of 7.32 µm diameter beads at the end of four different backflush events inside a square microfluidic channel. First quadrant with overlay to show membrane location and channel dimension. Membrane is visible in all frames where fluorescent bead irreversible fouling has occurred. Backflushed particle position is observed different in each event indicating new or different particles move in varying ways with each cycle.

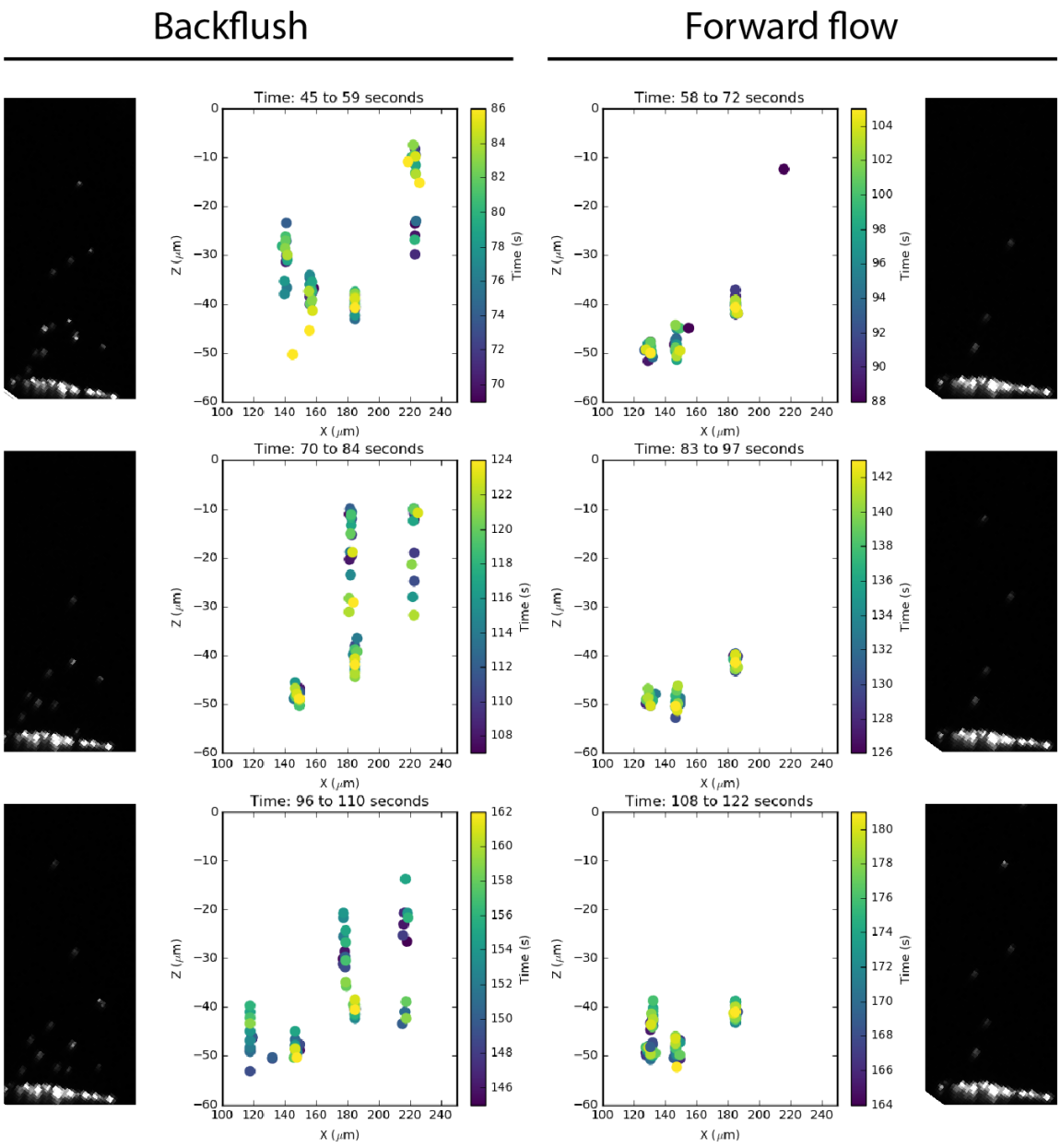


Figure 22 – Still frames and corresponding particle z-position over time for backflush and forward flow phases of successive cycles. Particles can be observed to express dramatic z-axis variation during backflush and not during forward flush suggesting backflush events create convective eddies within the yz-plane, reintegrating the cake with the bulk.

4.4.4 Modelling

In the fixed backflush experiments, we observed a marked change in the permeate output concentrations with decreasing duty cycle, resulting in an improved mean enrichment of target particles of 8.0-fold in comparison to forward flow. Recovery percentage is also critical to improve, and we showed a median value increase from 7.1% to 54.7% as duty cycle varied from 1.0 to 0.55 by implementing PWM flow control. To optimize the recovery percentage of the flow control system, we developed a mathematical model, Appendix A.5 that describes outcomes as a function of duty cycle through periodic cake disruption and re-integration into flow.

We assume a semi-permeable membrane that separate particulates suspended in fluid with a microfiltration process. Filtration through a porous membrane is governed by Darcy's law, in which permeate flux, *J,* is proportional to the differential pressure drop, *ΔP,* through a given thickness, *dz,* of a membrane as per Equation (39).[78]

$$J = \frac{1}{A_M}\frac{dV\{t\}}{dt} = -\frac{k}{\mu}\frac{\Delta P}{dz} \tag{39}$$

Darcy's law can be modified assuming a constant pressure drop across the membrane in which the right-hand side of Equation (39) can be simplified to:

$$J = \frac{1}{A_M}\frac{dV\{t\}}{dt} = \frac{\Delta P}{\mu(R_m + R_c)} \tag{40}$$

where R_m is the resistance due to the membrane and R_c, the time-dependent resistance of the filter cake [78].

A variety of fouling models for R_c are used to test whether pore-blockage, pore-constriction, or cake formation dominate at each stage in filtration. A model proposed by Ho and Zydeny[103] accounts for a transition between R_c models for protein fouling at all times through the dead-end filtration process as follows:

$$\mathrm{R_c} = \left(\mathrm{R_M} + \mathrm{R_{c,0}}\right)\sqrt{1 + \frac{\mathrm{f'R'\Delta PC_b}}{\mu\left(\mathrm{R_M} + \mathrm{R_{c,0}}\right)^2}\left(\mathrm{t} - \mathrm{t_p}\right)} - \mathrm{R_m} \tag{41}$$

This relation is then implemented into Equation (40) yielding Equation (42).

$$J = J_0\left[\exp\left(-\frac{\alpha\Delta PC_b}{\mu R_m}t\right) + \int_0^t \frac{\alpha\Delta PC_b}{\mu\left(R_m + R_p\right)} \times \exp\left(-\frac{\alpha\Delta PC_b}{\mu R_m}t_p\right)dt_p\right] \tag{42}$$

In the case of microfiltration of moderate solute concentration through a cellulose membrane, the solvent flux is assumed constant with time while the solute flux decreases exponentially to zero as t increases. This decrease in solute flux is the primary limitation in dead-end filtration[83,90,98]. We show that implementing a PWM periodic backflush cycle

to clear reversible fouling of the membrane from cake and pore can increase solute flux, recovery, and enrichment factor. The ratio of forward flow volume V_f to backflush volume V_r during a filtration operation is the duty cycle, ϕ: Equation (48).

$$V_f = V_f + V_r \tag{43}$$

A ϕ of 1, Equation (43), constitutes continuous forward flow with no backflush and is subject to standard exponential decay in flux as fouling occurs. Comparatively, $\phi = 0.50$,

$$V_f = \frac{V_f + V_r}{2} \tag{44}$$

results in the solvent volume dedicated to forward flow through the membrane equal to the solvent volume for backflush in any given periodic cycle and no net processing would occur.

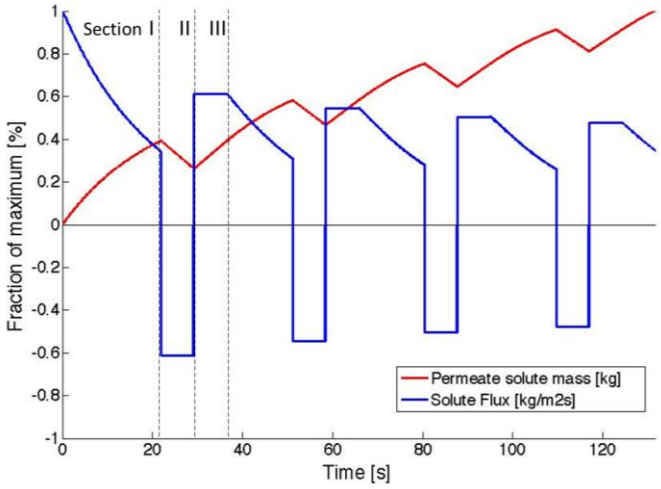


Figure 23 – Microfiltration model with periodic backflush, ϕ = 0.75. Flux profile in blue, permeate mass in red.

The profile of a microfiltration process with fixed periodic backflush volume is shown in Figure 23. With backflush volume fixed, increasing duty cycle result in increased total forward flow volume and decreasing cake thickness. Therefore, as ϕ increases, the cake thickness should grow proportionally. The forward flow volume is calculated as a function of ϕ and backflush volume per Equation (45):

$$V_f = V_r \frac{\phi}{1-\phi} \tag{45}$$

The process associated with forward flow (section I, Figure 23) results in an exponential decay period. The process associated with negative flux of Section II results in permeate particulate rescinding across the membrane during backflush. Section III represents a low resistance forward flux period immediately following backflush in which

the back-flushed permeate from Section II is pushed through the membrane again, after which, exponential decay again takes place. This cycle repeats for the entire volume of the sample processed. The benefit of this scheme is that the negative flux drastically changes forward flux resistance, resulting in a net improvement in forward flux. In this model, we assume an infinite bulk supply, keeping the exponential decay rate constant for each successive cycle, and incorporated irreversible fouling, resulting in sequential loss of flux capacity after each backflush cycle.

For each cycle, the model assumes a linear relation between duty cycle and the clearance of reversible fouling per Equation (46).

$$R_{c,n+1} = (1 - \phi)R_{c,n} \tag{46}$$

As ϕ varies from 1 to 0.50, a greater fraction of the reversible cake fouling is removed. Greater proportional fractions of backflush, i.e. as ϕ approaches 0.50, result in larger resuspensions of solutes from the cake into the bulk, increasing flux capacity for the system, and improving overall permeate recovery. Plotting permeate solute mass and solute flux as functions of time show increased instantaneous slope after backflush, Figure 23, resulting in monotonically increasing recovery percentage. Figure 24 explores the impact of ϕ on overall recovery as a function of time until 1 mL pf bulk has been processed. Figure 24 inset shows the 1 mL endpoint for each recovery percentage curve. The endpoints

increase to a maximum achievable recovery percentage of approximately 84.5% as ϕ approaches 0.50 determined by fitting an exponential growth curve to an asymptote.

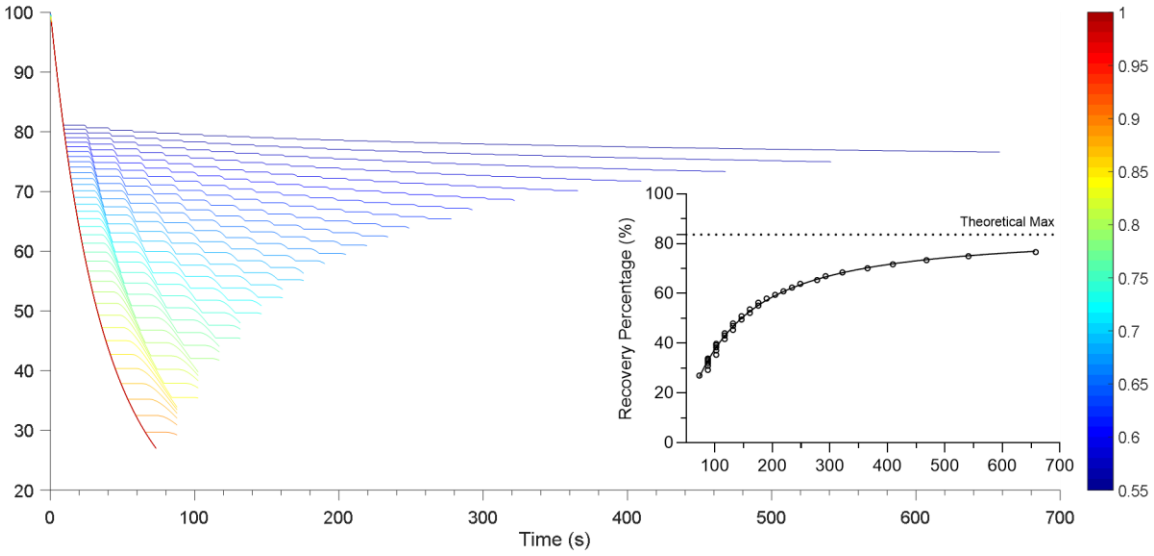


Figure 24 – A) Recovery Percentage as a function of time for ϕ from 0.55 to 1.0 in 0.01 increments. (Subgraph) End times for ϕ from 0.55 to 1.0 and calculation for maximum percent recoverable at ϕ = 0.50+. B) Theoretical maximum for recovery percentage as a function of ϕ showing linear gain with a maximum at y-intercept (ϕ = 0.5) compared to processing time vs ϕ.

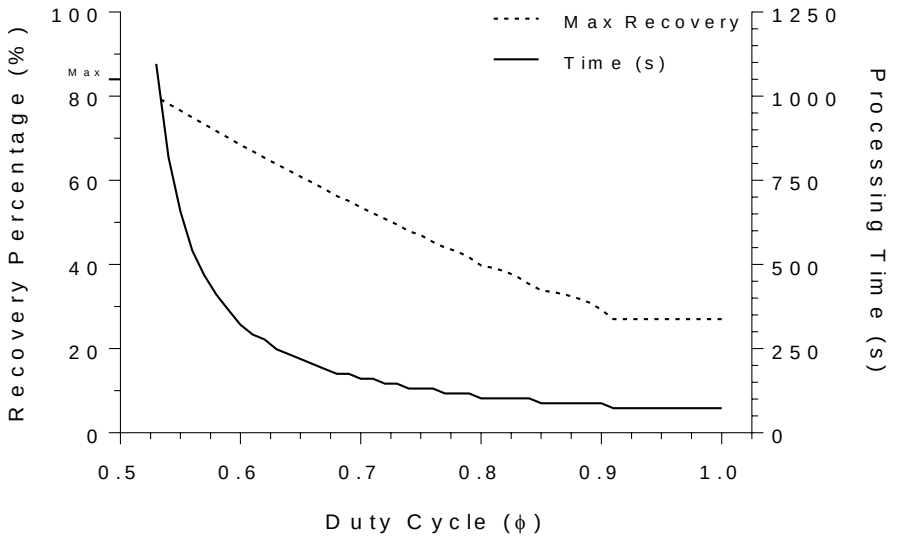


Figure 25 – Processing time and recovery percentage compared together as functions of duty cycle.

One consequence of this approach to maximize recovery is the increase in processing time, illustrated in Figure 25. As ϕ approaches the value of 0.50, the time to process a sample of bulk concentrate asymptotically approaches infinity. Additionally, we observed a critical duty cycle as ϕ nears 1.00 in which the calculated forward flush volume is greater than the total processed volume, which results in an effective ϕ of 1. Comparing percent recovery versus ϕ to percent recovery versus time, we can show a diminishing improvement of recovery per unit increase in process time, Figure 26. Further, we can calculate a maximum recovery percentage for each modeled duty cycle to see that the experimentally collected data closely follow the theoretical model and strongly correlates with a Spearman's correlation coefficient of 0.985. We have not observed an experimental

value that matches the theoretical maximum recovery percentage, Figure 27. The error between theoretical and experimental data suggests room for improvement of our device by increasing syringe precision to minimize spread and suggests improvements to the model by accounting for experimental losses from settling, adsorption, and human error.

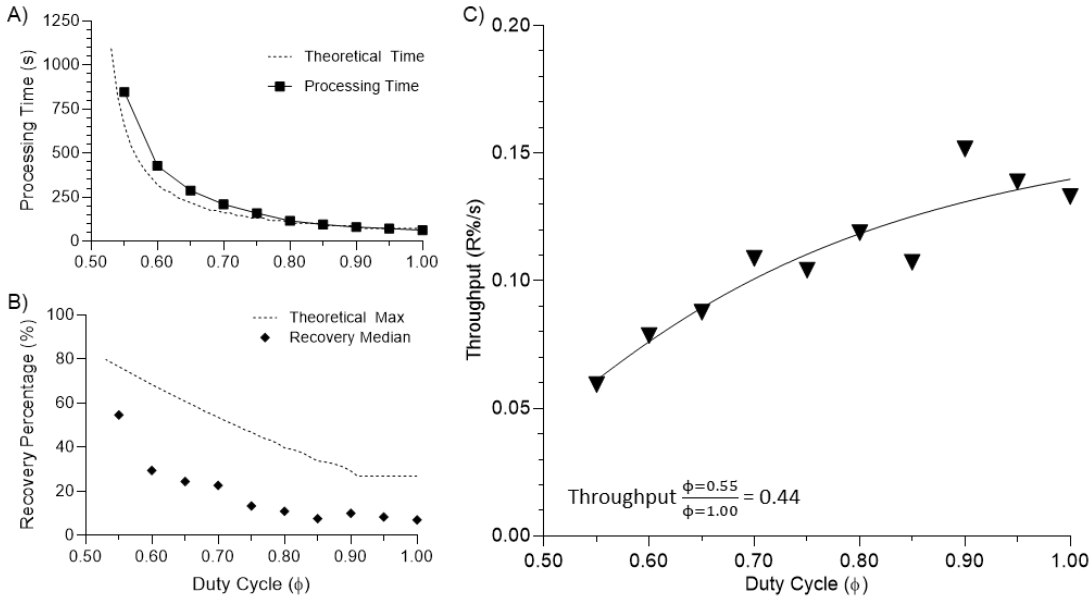


Figure 26 – A) Theoretical process time compared to observed process time as a function of ϕ. B) Theoretical maximum for recovery percentage as a function of ϕ compared to median recovery percentage. C) Throughput, calculated as recovery percentage divided by processing time, as a function of ϕ. Reduction of 44% throughput is observed as duty cycle decreases from 1.00 to 0.55.

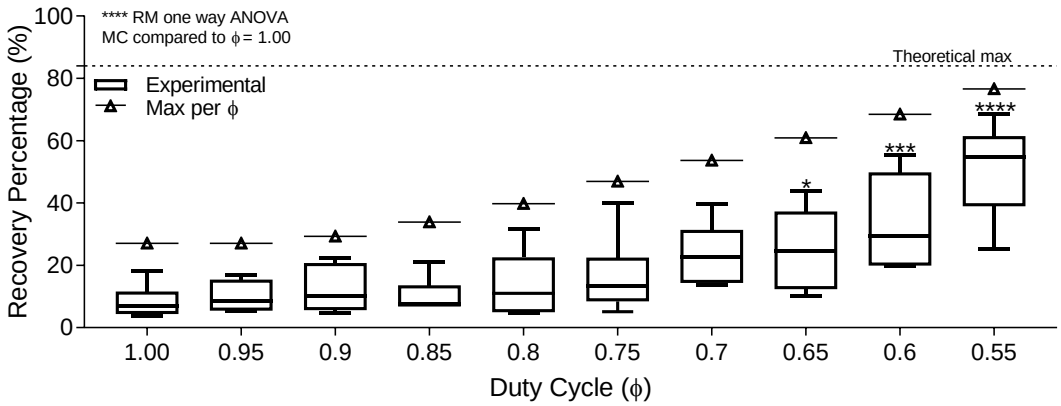


Figure 27 – Fixed Backflush Recovery percentage as a function of duty cycle compared to theoretical maxima at each duty cycle fraction, n = 10.

4.4.5 Mapping Expected System Performance

The model described in section 4.4.4 is only valid for fixed backflush PWM experiments. However, we observed a few changes to permeate output concentration that are not captured in our model, including stagnating recovery in fixed forward flush. Additionally, the initial model cannot analyze effects of pulse amplitude or frequency modulation. Further, although waveforms with rapid acceleration or deceleration produce the highest cake resistance reduction percentage[1], we wanted to confirm performance of alternate waveforms over a span of variables through the model.

The model generated previously uses only square waves for fixed backflush experiments. It was designed to calculate forward flush volumes as a function of duty cycle

and backflush volume. Calculating the waveform in this fashion limits the outputs to square waves of a single backflush volume. The model was modified to handle new input variables for multiple waveform packets. We developed a mathematical model, shown in Appendix A.6, that describes recovery percentage and associated processing time as a function of duty cycle through periodic cake disruption and re-integration into flow.

We assume a semi-permeable membrane that separates particulates suspended in fluid through the microfiltration process. Filtration through a fibrous membrane is governed by Darcy's law, in which permeate flux decays exponentially, but is distinctly separate from the solvent flux which can pass the membrane unattenuated. For the purposes of this model, we intentionally left the fouling and clearance rate constants as free variables within the system. Utilizing the data collected in FBF PWM experiments, we regressed the data against the medians and empirically derived these rate constants. Figure 28, shows the calculated recovery percentage for each stepwise duty cycle based on these regressed variables providing a fit of $R^2 = 0.953$ against the median and 0.972 again the mean.

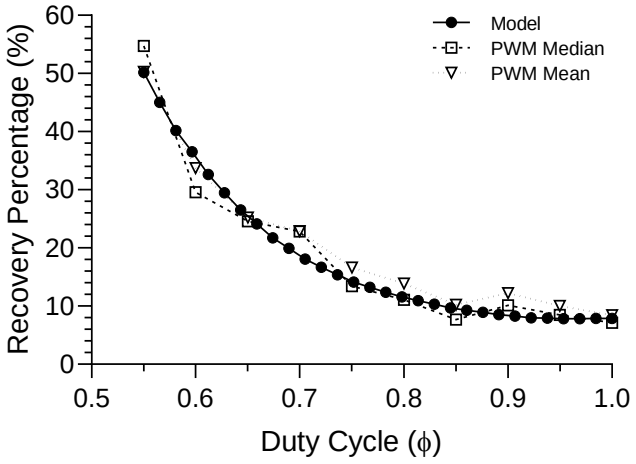


Figure 28 – Model calculated recovery percentage data acquired after regression of filter fouling and clearance constants were regressed against the median compared to the median and mean of experimental recovery percentage data from PWM FBF.

We use the refined model to calculate recovery percentage for a fixed volume of bulk solution processed given a fixed backflush volume and iterating through multiple duty cycles, Figure 29. By repeating this process for a range of backflush volumes and duty cycles, we create a 3D plot of all recovery percentage points possible. Figure 30, shows a representative surface generated. It is observed that highest recoveries occur at the lower duty cycles with lower backflush volumes, and that for a given duty cycle decreasing backflush volume produces diminishing returns on recovery percentage. Additionally, higher recovery percentage is shown to come at the cost of exponentially increasing processing time. The behavior reported herein parallels observations in both the FBF PWM and FFF PWM experimental sets. This map of duty cycles could theoretically be used to

select the optimal setting for a desired result: whether minimized time, maximized recovery percentage, low or high processing time, or more linear step response with duty cycle.

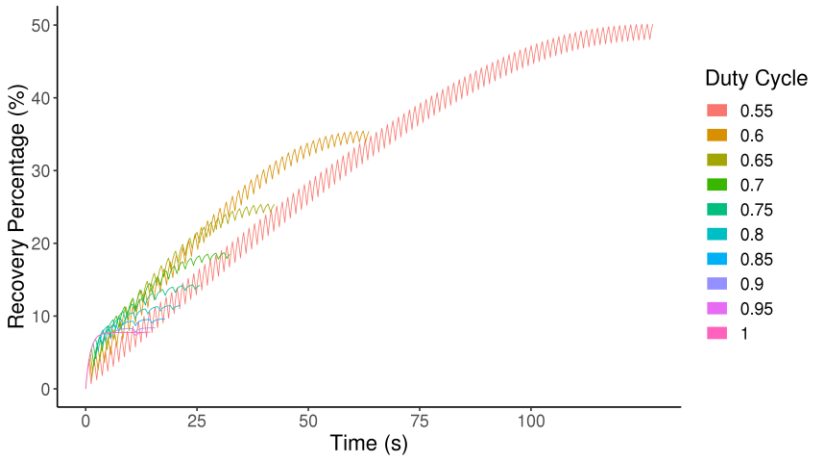


Figure 29 – Recovery percentage versus time for multiple duty cycles with fixed V_r.

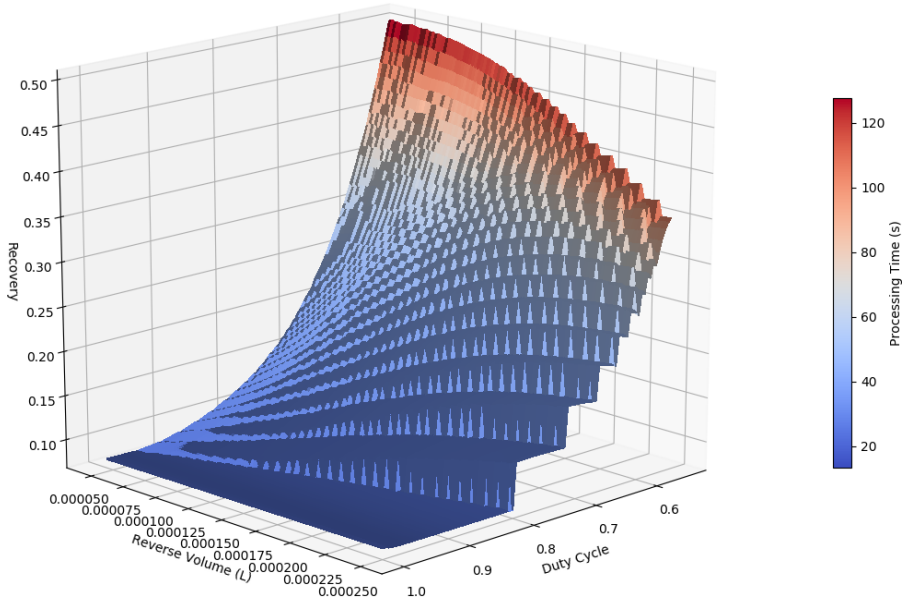


Figure 30 – 3D surface plot of recovery percentage with duty cycle from 0.55 to 1.00 and various backflush volumes. Colormap indicates theoretical processing time at each combination of independent variables allowing for selection of desired outcomes.

4.5 Discussion

We investigated the effects of pulse width modulated periodic backflush with variable duty cycle on cake formation interruption, fouling layer reintegration, and permeate flux improvement. Utilizing backflush volume, forward flow volume, and frequency as process parameters, we showed that PWM backflush cyclically restores flux capacity of dead-end filters. We developed a mathematical model to demonstrate that variation in the PWM duty cycle direct impacts recovery percentage and enrichment factor in dead-end systems, and that, experimentally, PWM backflush produces up to 18 times higher permeate concentration, significantly higher ($p < 0.05$) recovery percentages of 54.7% median compared to a baseline median of 7.1%, and significantly larger ($p < 0.01$) enrichment factor with an average fold change of 8.0, compared to constant flow rate. Overall, we show that higher PWM with smaller duty cycles paired with small backflush volumes produce the best recovery percentages.

However, improving recovery percentage comes at the cost of asymptotically increased processing time, and decreasing backflush volumes produce diminishing returns for recovery at the cost of throughput. To make the model more predictive, we refined it by regressing fouling and clearance rate parameters against the median of the collected FBF recovery percentage data. With more accurate model predictions, we can generate expected outcomes for a range of backflush volumes at any duty cycle. These predictions can be used to select desired outcomes prioritizing recovery or process time as needed.

CHAPTER 5. AMPLITUDE AND FREQUENCY MODULATION

5.1 Abstract

Pulse width modulated dead-end sample filtration was shown in CHAPTER 4 to improve recovery percentage over standard dead-end filtration techniques, but the method results in an asymptotically increasing processing time. We show the tradeoffs in throughput can be minimized with backflush algorithms applied through a customized fluidic actuator using variable flow rates and frequencies, significantly improving filtration efficiency. A Pulse Amplitude Modulation (PAM*)* process was used to periodically backflush the filter membrane, repeatedly interrupting cake formation while maintaining constant backflush volume at varying flow rates to reduce the time spent in the backflush phase. PAM flow control was applied to suspensions containing 2.19 um and 7.32 um polystyrene microbeads to produce an average 13.5-fold higher permeate concentration, with no significant decline for increased backflush flow rates. Further, pulse frequency modulation was applied to minimize flow direction time compensators and processing time. An optimal frequency of 1.24 Hz was shown to provide the largest recovery while minimizing processing time. Optimized via PAM and PFM, pulse modulated backflush was shown to reduce standard losses to throughput from 2.25x to 1.93x improving losses associated with a duty cycle of 0.55 by 16.8%.

5.2 Introduction

Investigations into membrane clearance using pulsatile flow for industrial applications focus on minimizing backflush volume while maximizing solvent flux through modulation of frequency and duty cycle[32,82,83,88–91]. These groups identify a range of duty cycle from 0.85 to 1.00 centered around 0.90 and a frequency range from 0.00001 to 0.01 as ideal for minimizing backflush volume, Figure 6. However, the investigations we performed in Chapter 4.4 have shown that minimizing backflush volume may not be ideal for maximizing recovery percentage of filtrate for dead-end systems, with duty cycles lower than 0.85 producing significantly larger results[104]. Unfortunately, although lower duty cycle percentages produce better recovery, this comes at a cost of about 2.25x lower throughput measured as recovery percentage per second.

$$\text{Throughput} = \frac{\text{Recovery Percentage}}{\text{Processing Time (s)}} \tag{47}$$

In Chapter 4.4.4, we show that we can control the disruption of cake and reintegration of the fouling layer to improve recovery percentage at the cost of increased processing time. We hypothesize that we can maintain these same improvements but minimize processing time by increasing frequency to decrease membrane contact time and direction change compensators, and fluid flow rate, amplitude. A schematic of the procedure is shown in Figure 31. The flow is controlled through pulse amplitude modulation (PAM) and pulse frequency modulation (PFM) of fluidic velocity wave vectors. We show the use of PAM and PFM periodic backflush compared to a uniform

forward flow and the results of PWM backflush from prior experimental sets. These experimental modalities allow us to explore the effects of cake clearance and backflush volume on overall system throughput. PAM applied to dead-end filtration produced over 13-fold higher permeate concentration compared to forward flow alone and maintained significantly consistent recovery percentage regardless of backflush velocity, and increased enrichment of 8-fold. PFM applied to dead-end filtration produced optimal recovery with 72.6% reduction in velocity direction change compensators.

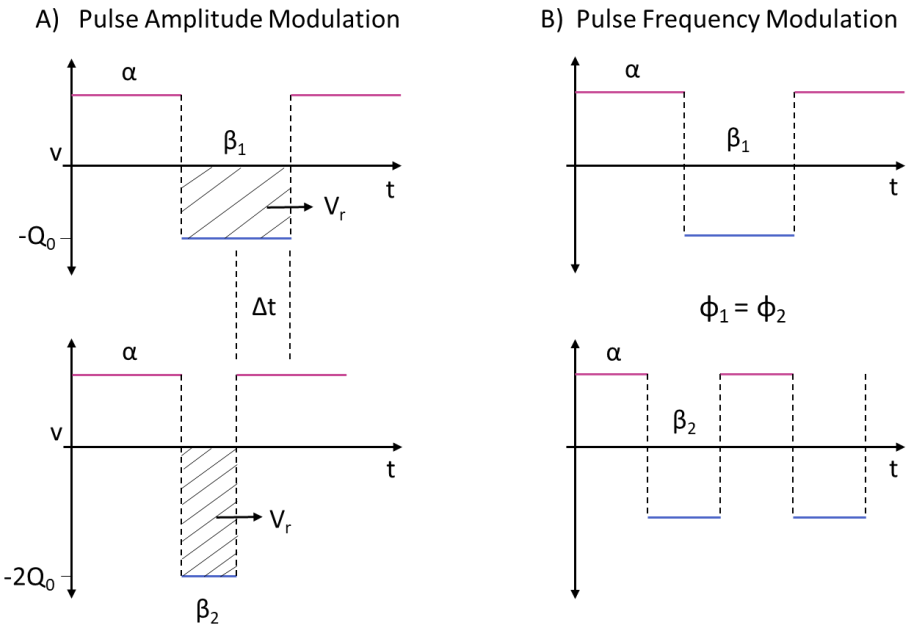


Figure 31 – Wave manipulation for pulse amplitude and pulse frequency modulation. A) Pulse amplitude modulation maintains constant backflush volume by changing backflush velocity and β, ultimately decreasing the period. B) Pulse frequency modulation maintains constant duty cycle holding the proportion of α and β constant and modulating frequency.

5.3 Methods

5.3.1 Membranes

The experiments were conducted with Pall Acrodisc syringe filters with Versapor (hydrophilic polypropylene). The filters are commonly used for both aqueous and organic samples with a glass-fiber prefiltration component. All filters have a diameter of 25 mm and functional cross-sectional area of approximately 2.8 cm^2.

5.3.2 Particulate sample processing

Mixtures of polystyrene (PS) microspheres were suspended in phosphate buffered saline solution with 0.02% tween by volume in PBS and were transferred to the system for processing via syringe. The particles included mixtures of 2.19 µm and 7.32 µm average diameter polystyrene particles (Bangs Labs) in a 50:50 ratio at a concentration of five million particles/mL, Figure 15. Particles were flowed through the filter using a syringe pump. PAM ad PFM experiments processed 1 mL of sample fluid at an absolute forward flow rate of 4.55 mL/min. For each experiment, the testing rig is primed with DI water prior to testing to eliminate bubbles and reduce capacitive actuation effects.

5.3.3 Volumetric Normalization

After processing the total volume of microsphere suspension samples through the membrane using PAM or PFM, all samples were normalized for volumetric differences in the priming stage arising from minor variations in the negative space of the stopcocks and filters using the following procedure. The output from both permeate and retentate were

collected in 5 mL culture tubes and centrifuged at 3500g for 15 minutes. The supernatant was removed, and the particle pellets were resuspended in 3.4 mL of DI water. All samples were then vigorously vortexed and kept on a tube rotator to break up the pellet and ensure uniform dispersion through the solution. The particle count for each condition was then measured using flow cytometry.

5.4 Results

5.4.1 Pulse Amplitude Modulation

PAM volumetric flow was conducted to hold backflush volume constant and independently examine multiple backflush volumetric flow rates. Figure 31 shows that in order to maintain backflush volume, proportionally smaller backflush durations must be maintained for increases in volumetric flow rate. Prior literature indicates that faster flow velocity rates of change in cross-flow systems result in higher cake resistance reduction leading to improved flux enhancement by increasing membrane shear forces[1]. The research shows a direct correlation between rate of change and particle detachment.

We hypothesized that, for dead-end systems, cake resistance reduction should also be affected by flow velocity rate of change. However, for square and pulse waves, recovery percentage should not be affected as rate of change of fluid flow is effectively instantaneous, or maintains the same rate of change, regardless of magnitude limiting clearance to a function of backflush volume[104]. By increasing the backflush volumetric

flow rate, we should be able to minimize the time spent during backflush and maintain high recovery percentage and reduce processing time. We therefore designed PAM periodic backflush experiments in which we modify the volumetric flow rate for fixed duty cycle ensuring constant backflush volume.

The amplitude of the backflush flow rate was found to maintain similar target particle concentration gain at the permeate side of the filter regardless of backflush velocity. Earlier, ϕ was defined as forward flow volume divided by the gross volume exchange through the filter or the sum of forward flow plus back flow volumes, Equation (48). Figure 32 shows the concentration gain and recovery percentage of smaller 2.19 μm particles from larger 7.32 μm particles initially mixed together and processed by the filtration system described in Figure 12, for increasing backflush flow rates while maintaining duty cycle, ϕ. Increasing the reverse flow rate while holding backflush volume constant allows us to diminish backflush duration by as much as 50% by doubling the backflush flow rate compared to the forward flow rate.

Particle counts from individual experiments were used to calculate a concentration fold change curve. The mean particle concentration for the control group with no backflush was used to calculate the concentration fold change as a function of flow rate, shown in Figure 32A. We find an average fold change improvement of 13.4 across all backflush flow rates with magnitude greater than zero.

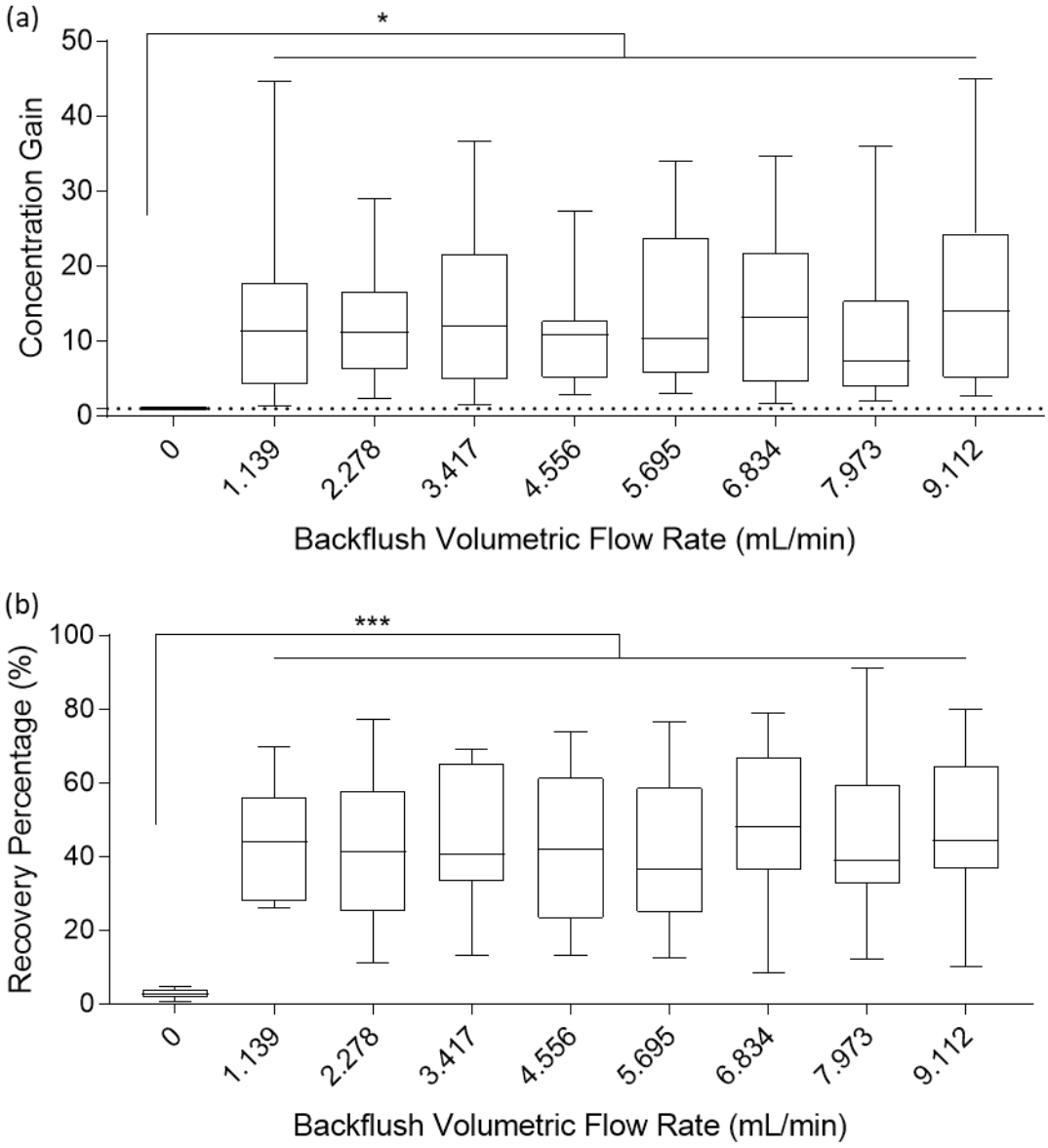


Figure 32 – A) Concentration gain and (b) recovery percentage as functions of backflush volumetric flow rate for fixed backflush volumes. No significant difference was observed between any samples with backflush flow rates greater than zero.

Recovery percentage, Figure 32B, shows the number of desired or targeted particles collected at the output compared to the number of desired particles initially supplied at the input, Equation (3). To compare against prior work, recovery percentage was scaled based on the ratio of actual to anticipated concentration fold change, normalizing against the concentration gain of the system. The recovery percentage significantly increased (one-way ANOVA compared to a control with $p < 0.0001$) from a median of 2.74% to an average median of 41.95%, for backflush flow rate equal to zero and the average median value for all flow rates with magnitude greater than zero. We found that recovery percentage in Figure 32B shows no significant difference between all backflush velocities when holding backflush volume constant.

The full range of backflush flow rates achievable by our syringe pump actuator was explored for an actuating syringe of volume 3 mL, from 0 to 9.11 mL/min, in linearly spaced intervals of 1.14 mL/min. For the control with no backflush, we observe a total processing time of 31.61 seconds at a flow rate of 4.556 mL/min. For all experiments testing backflush velocity variation, a compensator of one second was added between the forward flow and backflush regions for the flow to reach steady state. As a result, we observe monotonically decreasing processing time as the magnitude of backflush flow rate increases, Figure 33. From this relationship, we calculate a minimum achievable processing time to be 687.38 seconds in a system with infinite backflush flow rate. The difference between the processing time at the standard processing rate of 4.556 mL/min and the theoretical minimum processing time is 141.84 seconds, assuming no loss of

recovery and an infinite backflush flow rate. By doubling the flow rate to 9.112 mL/min, we observe a processing time of 758.30 seconds, cutting the difference observed before in half. The difference between processing time and minimum theoretical will continue to decrease by a factor of two for every doubling flow rate, ultimately diminishing the returns for high backflush flow rates.

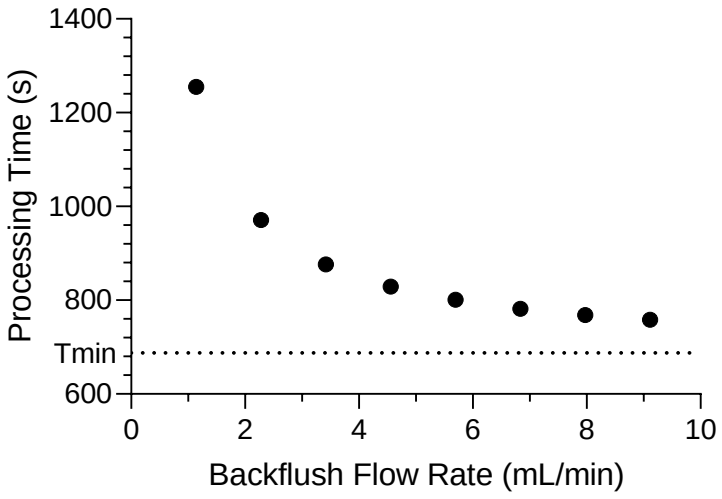


Figure 33 – Processing time as a function of backflush flow rate. Processing time will asymptotically approach infinity as backflush flow rate approaches zero, but a backflush flow rate of zero results in only forward flow rate determining processing time calculated as volume divided by flow rate. A minimum processing time of 687.38 is achieved for infinite backflush flow rate.

5.4.2 Pulse Frequency Modulation

The system uses a compensator, δ, to delay switching from forward flow to backflush and vice versa to allow particulate mass to approach a movement steady state by allowing pressure differentials to equalize. This compensator was set to one second in

duration, determined experimentally, for all PAM experiments. As frequency decreases for the same duty cycle, so too should the compensator, Figure 34, as less fluid is being moved and pressure differentials are equalized and reversed sooner. Here, we propose a model in which δ is proportional to the sum of α and β.

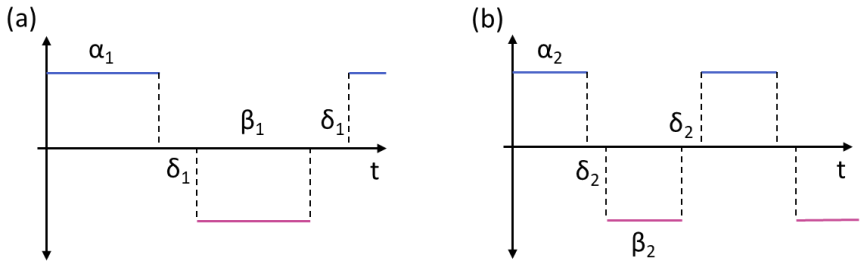


Figure 34 – Two square waves with different frequencies but equal duty cycles. The primary difference between (a) and (b) is the time contribution of the compensator δ with $\delta_1 > \delta_2$.

For basal experimental design set at a duty cycle of 0.55, a flow rate of 4.556 mL/min, and a gross volume exchange of 0.0933 mL on the device described in Figure 12, a compensator of one second was experimentally found to be most effective. We use this as the reference point for calculating the change in δ proportional to the change in frequency of the wave. The system executing backflush does so in discrete volumetric steps. We hypothesize that we can leverage the deviations of frequency from this discrete approximation of duty cycle to minimize processing time. Further, decreasing δ be reducing backflush and forward flow volumes will reduce cake layer formation and irreversible binding, maximizing recovery percentage.

As seen in Figure 35, δ can be expressed as an inverse function of frequency. Additionally, processing time follows a periodic sawtooth as frequency increases. The sawtooth curve was shown to decrease processing time by 30.64 seconds for each unit increase in frequency. Extending this plot to investigate frequencies up to 10.6 Hz we can see a processing time reduction of up to 70% with a minimum processing time of about 250 seconds.

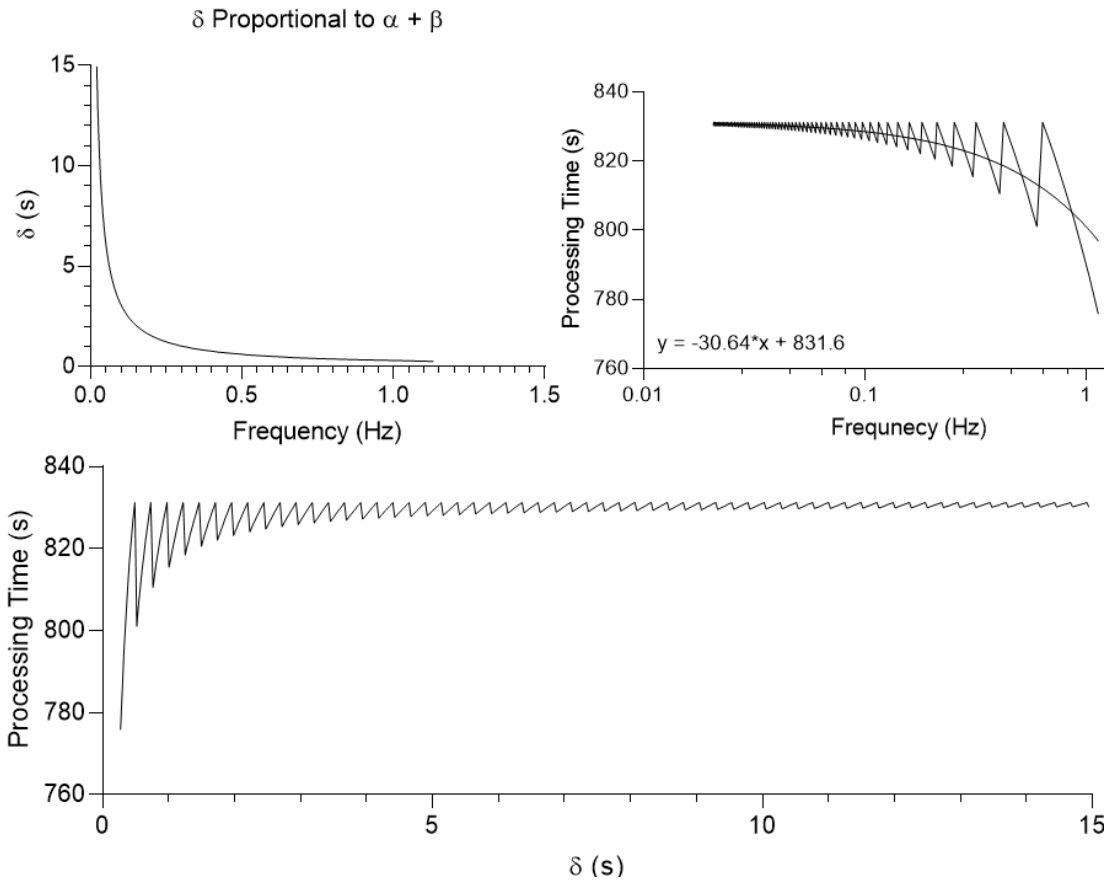


Figure 35 – Upper left: δ as a function of frequency. Upper right: processing time as a function of frequency on a logarithmic x axis. Linear fit equation provided in lower left corner of figure. Lower: processing time as a function of δ.

A more pronounced trend is observable when calculating the real duty cycle for this system. Because the system is actuated via stepper motor, the best approximation for a duty cycle of 0.55 is calculated given a backflush volume and provided duty cycle. A closest steps backflush is determined followed by a rounded approximate for steps forward flow in discrete integer values. In Figure 36, we show the true value duty cycle calculated from whole integer step resolution of our fluidic actuator. As frequency increases, total steps forward and backward are reduced, and rounding error in duty cycle increases. Plotting total processing time against these duty cycle variations shows a linear decline in processing time as duty cycle increases.

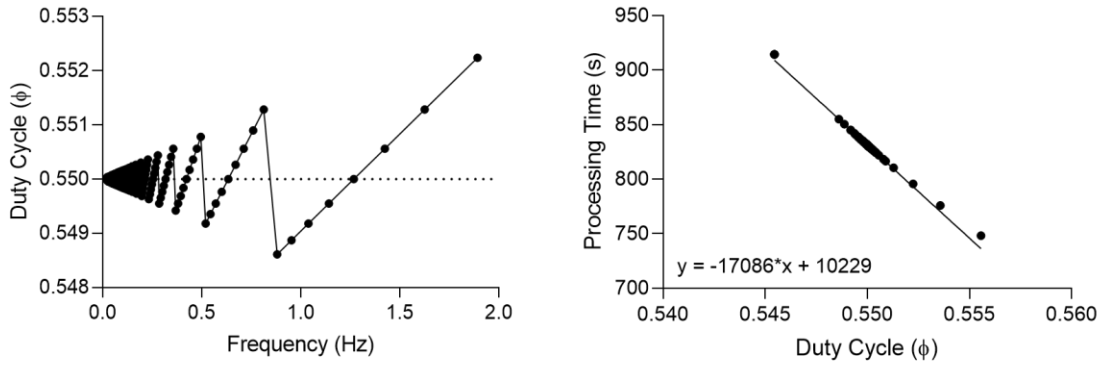


Figure 36 – Approximation of duty cycle as a function of frequency given single step resolution of 0.042 mL and associated effect on processing time as a function of duty cycle.

We use this to select the closest duty cycle approximation and frequency that minimizes processing time and ensures a viable execution combination in the system, shown in Figure 37. The highest frequency selection in this system provides the lowest

processing time and allows us to maximize for throughput. Compared to a duty cycle of 0.55 at a frequency of 0.31 Hz, we can reduce total processing time by 53. 38 seconds or about 6.4% using PFM alone. This would result in an increase in throughput of 6.84% on average.

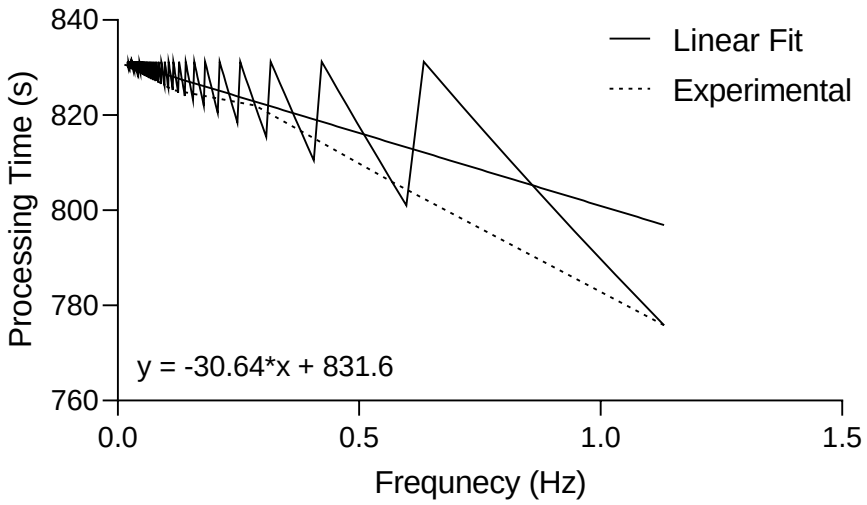


Figure 37 – Frequency vs Processing Time experimental selections to maximize throughput for duty cycles approximately equal to 0.55.

5.4.3 Improving Throughput

We can use the minimized processing time from PAM and PFM to understand the change in throughput for the system. Both amplitude and frequency changes can be reported through frequency. By decreasing the processing time through amplitude and frequency modulation, we can show throughput gains as a function of frequency for each system. Figure 38 shows these changes in throughput. We observed an improvement of

38.5% and 75.8% for PFM and PAM respectively when comparing the poorest performing metric to the best independently. When compared to the standard observed in PWM fixed backflush experiments, we observe a 15.8% and 27.6% improvement to throughput for PFM and PAM respectively.

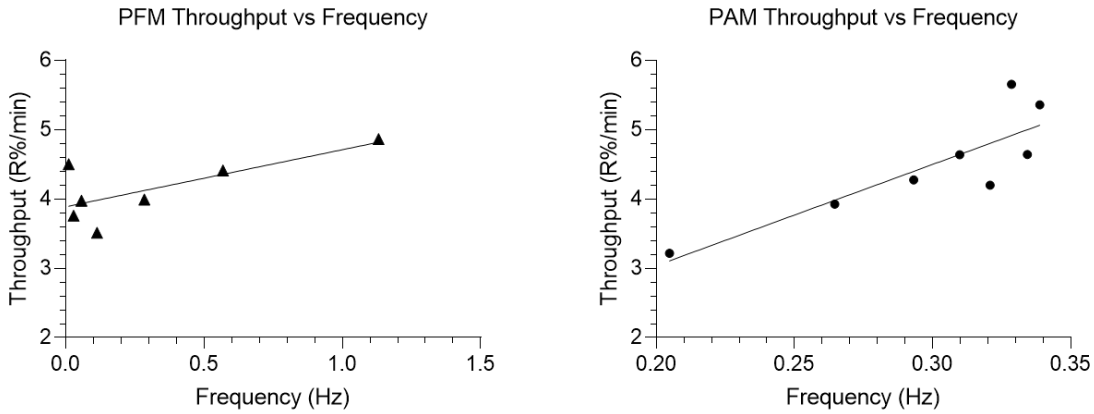


Figure 38 – Throughput as a function for frequency for pulse frequency and amplitude modulation

Combining the effects of PFM and PAM, we can decrease throughput further. We observe a median throughput increase of 38.8% with bounds of 5.6% and 85.7% for the combined effect of PAM and PFM. The improvement to throughput brings the median throughput to 61.9% of the maximum systematic throughput achieved at a duty cycle of 1.00; an increase from the original reported 44.4%. Figure 39, compares the individual improvements associated with PFM and PAM to the combined effects and original reported throughput value. We then apply these processing times to experimental throughput values

and show significant improvement between experimentally achieved throughputs, Figure 40.

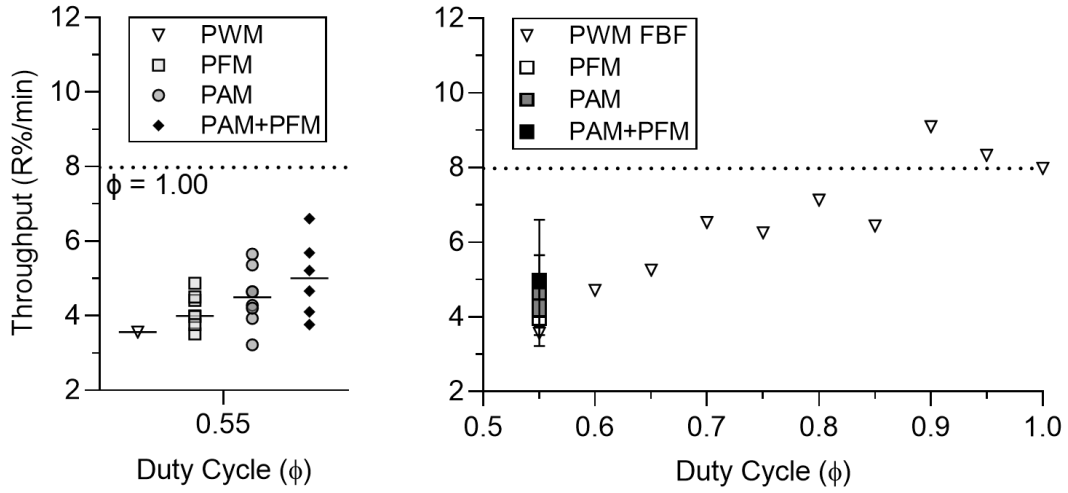


Figure 39 – Throughput gain as a function of duty cycle compared against pulse width modulated periodic backflush throughput for frequency modulation, amplitude modulation, and combined effective modulation. Each point shows the improved throughput calculated as the recovery percentage median from PWM for the new processing times of each experimental subset. Right graph shows ranges compared against median values from PWM experiments.

To compare the gain in experimental and theoretical throughputs against the values observed for duty cycle equal to 1.00, we use percent error to evaluate differences.

$$\text{Percent Error} = \left(\frac{Throughput - Throughput_{\phi=1.00}}{Throughput_{\phi=1.00}}\right) * 100 \qquad (48)$$

Using the equation above, we find that initial losses in throughput using the median value from the PWM experiments result in a percent error against ϕ = 1.00 of 50.4%.

Theoretical PAM processing time produces a percent error of 25.1%, PFM produces an error of 31.1%, and combining the two results in a total error of 15.1% providing a 35.3 percentile point increase. When testing experimentally, Figure 40, we see a percent error of 32.1% for PAM, 33.6% for PFM, 27.0% for the combination of PAM and PFM resulting in a 23.4 percentile point improvement. This implies that there would be room for improvement within the system resulting in an increase of 11.9 percentile points of error.

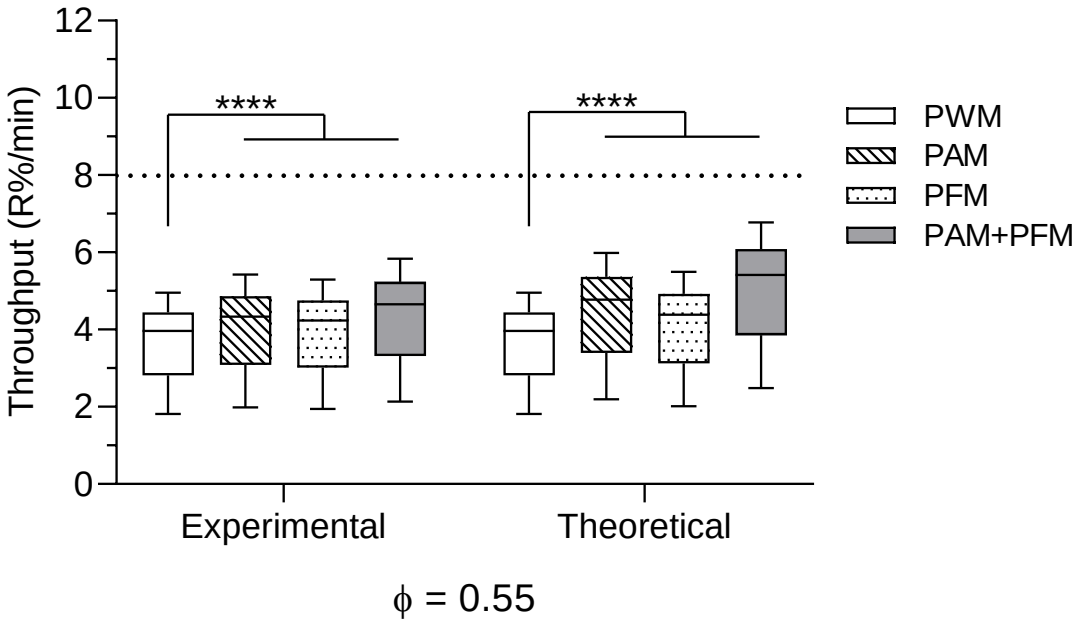


Figure 40 – Experimental values of improved throughput using PAM, PFM, and a combination of PAM and PFM to reduce processing time. Theoretical calculations show room for significant improvement beyond experimentally achieved results.

5.5 Discussion

We investigate the applications of PAM and PFM to improve the throughput of PWM periodic backflush systems. Compared to a duty cycle of 0.55 at a frequency of 0.31 Hz, we show a theoretical reduction in total processing time by 53.38 seconds, or 6.4%, for a frequency of 1.13 Hz in PFM experiments. Results from PAM show a theoretical reduction of 70.92 seconds, resulting in a processing time reduction of 8.5% with a theoretical maximum reduction of 141.84 seconds. The observed improvements associated with PFM and PAM independently are compounded when used together, resulting in median improvement of throughput by 38.8% with a maximum observed improvement of 85.7% compared to baseline PWM experiments. The combination of amplitude and frequency modulation can be used to increase throughput of pulse width modulated periodic backflush flow control reducing total error when compared against the median throughput for $\phi = 1.00$ by 23.4 percentile points and theoretically minimizing throughput error to within 15.1% of that achieved by $\phi = 1.00$.

CHAPTER 6. APPLICATIONS

As discussed in Chapter 2.4, sample preparation, tissue processing, and laboratory automation account for more than $11 billion in today's market. CHAPTER 4 discusses how the yield from particle separation processing can be improved, and CHAPTER 5 discusses how we can optimize this gain in recovery while maintaining throughput. These chapters represent the discovery and optimization of a pulse modulated technique to improve cell and particle isolation. Here, we investigate how we can apply this technology to improve specific applications. The purpose of this chapter is to investigate applications of the technology for conjugated microsphere recovery, cellular separations of bacteria and human cells, and tissue engineering through improved scaffold seeding.

6.1 Improving Manufacturing of Immune Engineering Particles

6.1.1 Rationale and Significance

Microparticles are used in therapeutics and laboratories for isolation, identification, treatment, and immune-engineering. Filtration of microparticles is frequently used to generate high purity particles for diagnostics and therapeutics[51,105]. Even non-degradable microparticles are useful in-vitro to study biological phenomena. Pacheco et al, for example, demonstrated that for IgG-functionalized polystyrene particles, classical complement pathway activation increases commensurate with Fc density and inversely with particle size[106], while macrophage uptake for smaller Fc-coated particles (1 μm or

less) was found to be related to the extent of IgG coverage[105]. These insights have informed the design of vehicles to develop host-modulated antibacterial approaches[53] and helped maximize bioactive payload delivery to macrophages or tune their downstream activity.

Additionally, microparticles are used in protein isolations as well. These techniques are focused on the isolation of cellular components[56–58] through adsorption to functionalized particles or antibody collection on the surface of microparticles directly[59,60]. To achieve separation and collection with microparticles, the microparticles are added to suspensions, incubated to allow time for adsorption and binding, filtered and unbound.

There are numerous techniques that make up a multibillion dollar industry for separations in pharmaceuticals and biotechnology[54,55]. For these applications and others, it is critical in the manufacturing process to generate high purity of targeted microspheres, but this process may not be pursued if production is not sufficiently profitable. Considering profitability restrictions can limit access to or availability of these applications, it is significant to demonstrate that we can apply pulse modulated periodic backflush to the recovery of conjugated microspheres to improve yield after protein conjugation protocols.

6.1.2 Methods

Silica microparticles, 0.96 μm in diameter (Bangs Laboratories, Fishers, IN), were coated in 5 nm chromium and 20 nm overlaying gold for functionalization as described by Tang and Holt, et al[107,108]. The particles were filtered in fixed backflush experiments to

confirm that beads designed for immunoengineering experiments could be efficiently purified[109]. Prior to experimentation these suspensions were washed and suspended in DI water by centrifugation (3500g for 15 minutes) to remove buffers and preservatives. The sample bulk is run through a cell strainer pre-filter stage and then either processed by pushing the bulk through a membrane at $\phi = 0.55$ or $\phi = 1.00$. Functionalized particles were used in volumes up to 10 mL to demonstrate applicability of PM periodic backflush within laboratory particle processing applications.

6.1.3 Results

We show periodic backflush can be used to streamline the functionalization process of silica microparticles, 0.96 μm in average diameter, and to improve purity and yield by removing aggregate particulates that results from the functionalization process through filter flux capacity restoration via PM periodic backflush. All debris was observed to be cleared from the functionalized bead population, Figure 41, and PM periodic backflush with $\phi = 0.55$ provided purity equivalent to $\phi = 1.00$ with significantly higher particle recovery percentage, ($p<0.001$), up to 3-fold greater than what was observed in standard processing applications, Figure 41A. Further, enrichment did not see significant gain compared to the base case, $\phi = 1.00$. Enrichment fold-change enhancement is expected if only targeted particle concentration increases in the output, Equation (2). As can be seen in Figure 41 D and E, although the quantity of particles per bin increases from an average of 15,000 to over 50,000, the ratio of targeted to untargeted particles in the output remains

approximately the same at 77.4% to 22.6% for the case of ϕ = 1.00 and 88.0% to 12.0% for the ϕ = 0.55 case. As a result, we can determine that the purity remains approximately unchanged and PM for these particles increases yield of both targeted and untargeted particles, indicating that filter exclusion may not provide a tight enough band.

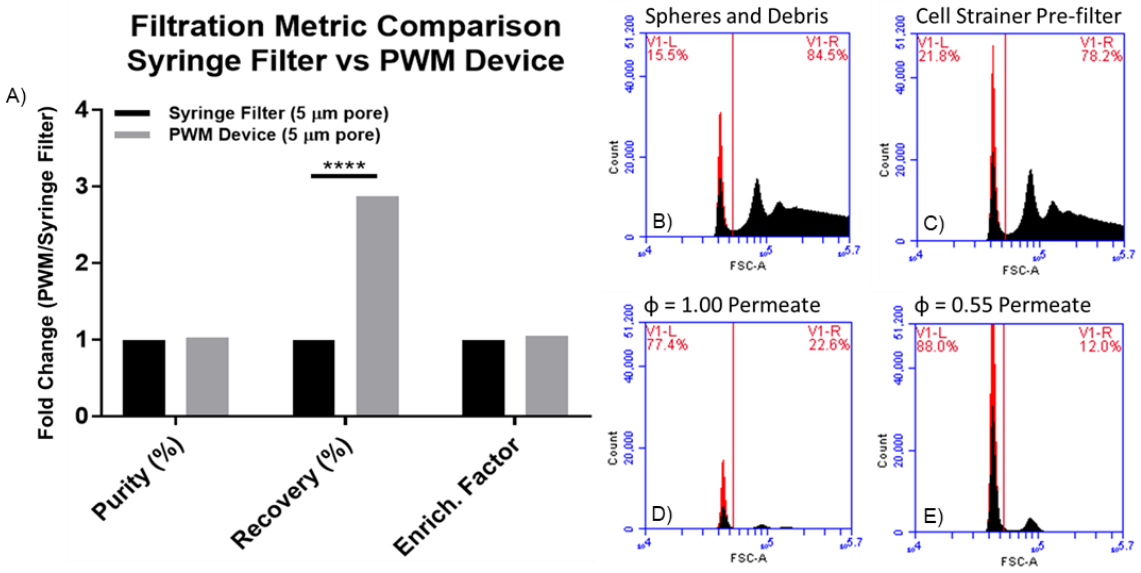


Figure 41 – A) Purity, recovery percentage, and enrichment factor for conjugated microspheres. ϕ = 1.00 compared to ϕ = 0.55 normalized to ϕ = 1.00. B) Conjugated microsphere forward scatter immediately following conjugation, and C) representative population after conjugation and pass through large pore cell strainer. D) Representative population of (B) passed through dead-end filter at ϕ = 1.00. High loss is apparent, but target sample purity is high. E) Representative population of (B) passed through dead-end filter at ϕ = 0.55. Purity is maintained in a similar proportion to that in (D), but population count is, on average, 2.88 times higher.

6.1.4 Discussion

Experimental investigations into conjugated microsphere separations provide a few critical insights from these experiments. PM backflush is a reliable option for not only

separating smaller particles than initially tested, 0.96 μm compared to 2.19, but also those made from various materials, silica compared to polystyrene. Additionally, we show from these experiments that we can apply PM to improve yield from uniformly distributed debris fields containing multiple irregularly shaped materials. We also show that filtration recovery percentage can be improved for functionalized particles without interfering with their effectiveness or losing significant quantities through prolonged contact with membranes potentially increasing losses via adsorption. We show a median of approximately 3x fold change in recovery percentage increasing yield from a median of 7% to greater than 21%.

6.2 Bacterial Separations from Mammalian Cell Culture

We have demonstrated that we can use PM periodic backflush to improve the permeate flux, recovery percentage, and enrichment of microspheres. Given that numerous clinical diagnostics rely on fractionated cell content, this experimental set will test if our PM periodic backflush system can be used to overcome the additional disadvantages associated with biofouling in clinical diagnostics. We propose experiments to validate the application of PM backflush on cells in clinically relevant examples. Here, we show that PM periodic backflush can be used to improve the recovery rate of bacteria in a cystic fibrosis diagnostic analogue.

6.2.1 Rationale and Significance

Cystic Fibrosis (CF) is a genetic disorder that results in the overproduction of mucous and chronic inflammation in the lungs. Chronically inflamed lung environments result in a state of recurring pulmonary infection from several infectious agents resulting in 80 to 95 percent of CF patients succumbing to respiratory failure[61]. Young patients are especially susceptible to organisms such as *Staphylococcus aureus* and *Haemophilus influenzae*, leading to compounding or long-term infection by *Pseudomonas aeruginosa*. It can be particularly important in CF infections to identify the infectious agents prior to treatment due to adverse effects resulting from toxin release from antibiotic and bacterial interactions[110].

Numerous studies have been conducted to determine the best way to identify lung infection in young children[111–114]. The care pathway for this diagnosis consists of onset of symptoms, presentation to hospital or urgent care center, collection of sample fluid from bronchoalveolar lavage (BAL), filtration of contaminating agents, cell culture to critical mass of cells, followed by genetic and bacterial analysis. Numerous studies have been conducted to show low accuracy of pathogenic bacterial identification resulting from this process, limiting the ability of clinicians to provide targeted care[111–115]. Low accuracy in identification can arise from many problems, such as oropharyngeal contamination or improper BAL sampling, but one important issue arises from the fact that BAL fluid samples are obtained from only a small portion of the lung, leaving the possibility that pathogens might be present in parts of the lung not sampled by lavage or that low recovery percentage in the filtration stage may result in removal of the pathogenic substance.

Bacterial recovery percentage improvement should directly translate to the time it takes to perform a CF diagnostic.

6.2.2 Culture Methods

Lung Epithelial Culture - Lung epithelial cells (a549 cell line) were transformed to produce GFP (sourced from Dr. Krishnendu Roy lab) and cultured in media consisting of 89% Roswell Park Memorial Institute (RPMI) base, 10% Fetal Bovine Serum (FBS), and 1% Penicillin-Streptomycin, Figure 42. Cells were maintained in a CO2 incubator at 37 degrees Celsius and passaged every three days at approximately 75% confluency. A hemocytometer was used to establish concentration measurements when suspended.

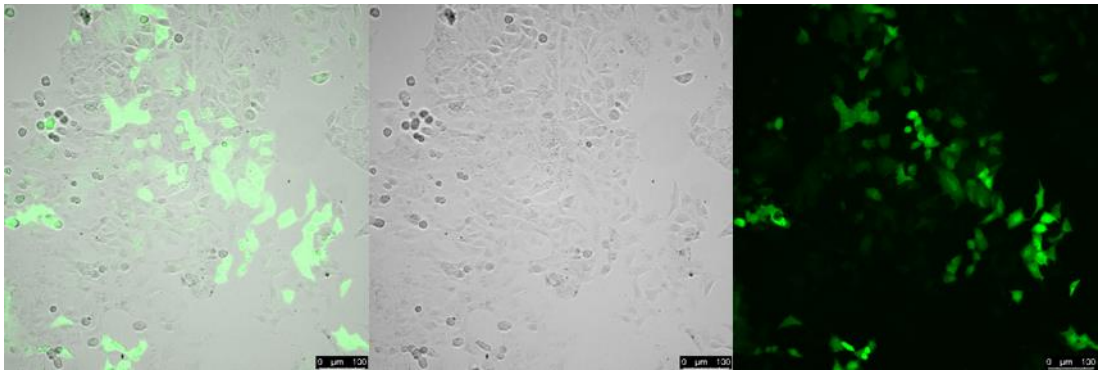


Figure 42 – A549 lung epithelial cell line culture scale bar 100 µm. (A) Overlay fluorescent and bright field image, (B) bright field only, and (C) fluorescent imaging only.

Bacterial Culture - Staphylococcus Epidermidis was cultivated in a conical tube with Luria Broth suspension media as an analogue for S. Aureus. When the S. Epidermidis suspension reached high turbidity, the bacterial solution was seeded on a LB agar petri

dish. After 24 hours, a colony was transferred to a new conical tube inoculating the media. This culture was mixed with the a549 to simulate a sputum sample

To determine colony forming units (CFU) per volume, a serial dilution of 0.1 mL of bacterial broth was performed resulting in 1:100, 1:10,000, and 1:1,000,000 dilutions. The dilutions were vortexed until uniform, and 10 µL from each were inoculated and evenly spread onto agar coated plates. The plates were incubated for 24 hours, and distinct, countable, formed colonies were summed from the plate. The original number of CFU per mL of solution was then derived by multiplying the CFU by the given dilution factor and averaged between plates of the same serial dilution line when possible.

6.2.3 Methods

Prior to experimentation, a CF sputum sample analogue is created. The analogue is a mixture of bacteria and lung epithelia in a ratio of 50:1 with a concentration of $5*10^6$ and $1x10^5$ cells/mL respectively. To achieve this suspension while maintaining cell lines, the CFU of the bacterial culture is determined 24 hours in advance of the experiments, and both cell lines are passaged immediately prior to experimentation. After trypsinization and reseeding of the a549 cell line, the remaining cells are washed to remove the antibiotic in the suspension media and resuspended in a neutral buoyancy buffer agent primarily composed of PBS and Percoll. The epithelial cell suspension is then spiked with the appropriate volume of the bacterial suspension to meet the ratio 1:50 ratio. This suspension is then tested immediately to minimize losses associated with the interaction of both cells.

Two microsieve filters, Whatman Nuclepore Track-Etch Membrane and Whatman Cyclopore Track Etched Membrane, were used to separate the cell-bacteria mixture. The filter membrane was encased in tight housing and connected to the customized syringe pump. Prior to loading the mixture into the feed stream reservoir tube, the system was primed with a sterile PBS and Percoll-based neutral buoyancy buffer to eliminate air bubbles from fluidic circuit. A volume of 1 mL of a549 spiked with *S. Epidermidis* was then processed for both filter types at $\phi = 1.0$ or 0.55. The permeate for all samples were collected, diluted to 1:10,000 and 1:100,000, and 10 µL of each dilution and the sputum analogue was plated uniformly on LB agar dishes using maracas-style glass bead spreading technique. The cultures were incubated for 24 hours prior to CFU counting. A center line tool was manufactured, Appendix B, and used to split dense plates into quadrants. A custom petri dish photo booth was constructed, Appendix B, to capture images of agar plates. Colony counts were performed manually using FiJi imaging software as is presented in Figure 43A. Manual colony counts were performed to confirm the results of the digital counting technique when feasible, Figure 43B.

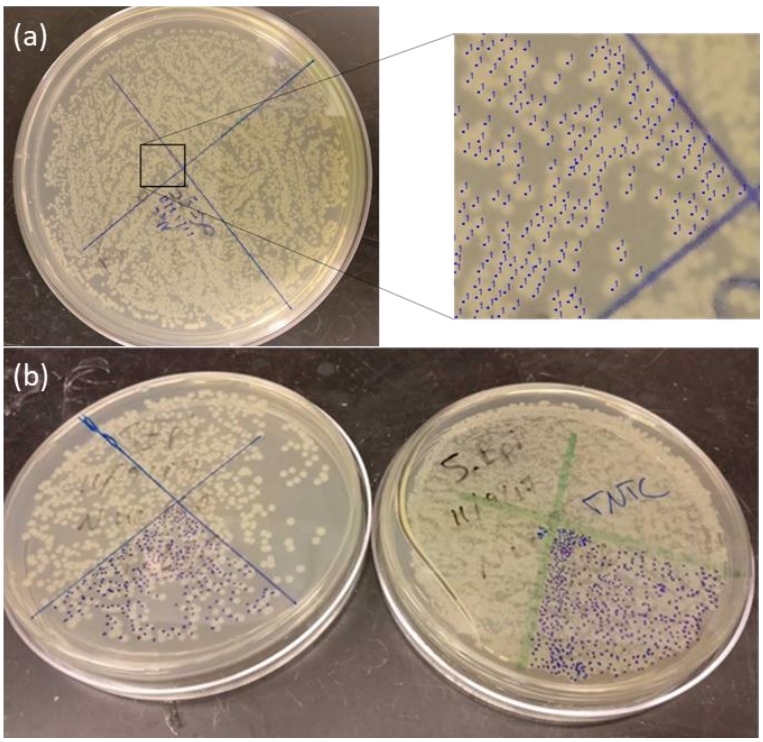


Figure 43 – Colony counting techniques for quadrant based CFU determination. A) FiJi (Image J) was used to determine a digital count and a B) manual technique for counting colonies.

6.2.4 Results

All count data was normalized to stock solution CFU providing a recovery percentage for bacterial load output from the system. Figure 44, shows that a duty cycle of 0.55 provides a significant increase in bacterial permeate, for both the Nuclepore ($p<0.01$) and Cyclopore ($p<0.001$) membranes. We see a median recovery percentage fold change of 1.88 and 2.31 respectively. The basal recovery percentage for $\phi = 1.00$ experiments showed a substantial recovery at 42.4%. As a result, recovery from Nuclepore and Cyclopore PM experiments are at a median of 79.7% and 97.94% respectively.

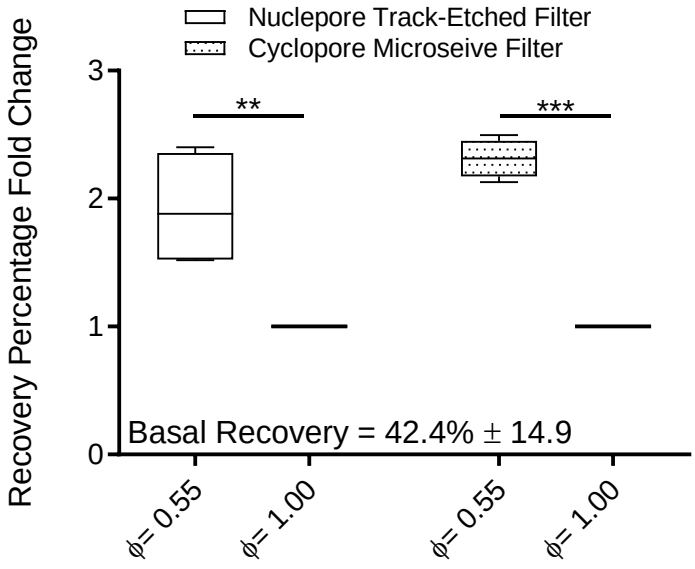


Figure 44 – Recovery percentage fold change for Nuclepore and Cyclopore filters processed at ϕ = 0.55 and ϕ = 1.00 normalized against ϕ = 1.00.

6.2.5 Discussion

We show the application of PM to bacterial filtration processes can produce an average median fold change of 2.1. We show an average median return of 88.8% when utilizing PM to improve the recovery of S. Epidermidis. The exceptionally large recovery percentage is likely attributable to the low concentration of bacteria in the suspension, their small size compared to the filter pores, and the large filtration surface area relative to volume processed. In future studies, we believe we can use the PM periodic backflush system to fractionate other cell types and even blood constituents to collect mononucleated and polynucleated leukocytes from diluted blood without additives such as Ficoll-Paque.

6.3 Scaffold Seeding

6.3.1 Rationale and Significance

Large deficits resulting from trauma, surgery, or slow or non-healing defects can be found in bone, cartilage, tendon, and ligament tissues resulting in more than one million orthopaedic repair surgeries per annum and costs the U.S. more than $5 billion[64,116]. Recent advancements in tissue engineering have enabled the use of perfusion bioreactors to grow autograft replacements for these repairs[71]. This minimizes downstream costs and reduces graft rejection in patients. Although perfusion bioreactor seeding studies have sought to improve uniformity and seed density in osteogenic grafts[67,72], this study will be the first to integrate in-line flow control to increase scaffold penetration in perfusion bioreactors. Thus, PM periodic backflush provides an appropriate means to improve penetration, uniformity, and seed density in scaffolds.

6.3.2 Methods

6.3.2.1 Decellularization

For these experiments, two types of scaffold were used for seeding: decellularized murine tibial trabecular meshwork (sourced from Dr. Robert Guldberg lab) and collagen sponge (sourced from Dr. Nick Willett lab). Murine trabecular meshwork was sourced from sacked but previously healthy rats and decellularized. The decellularization process required incubation in Triton X-100 on a rocker plate for 24 hours[67].

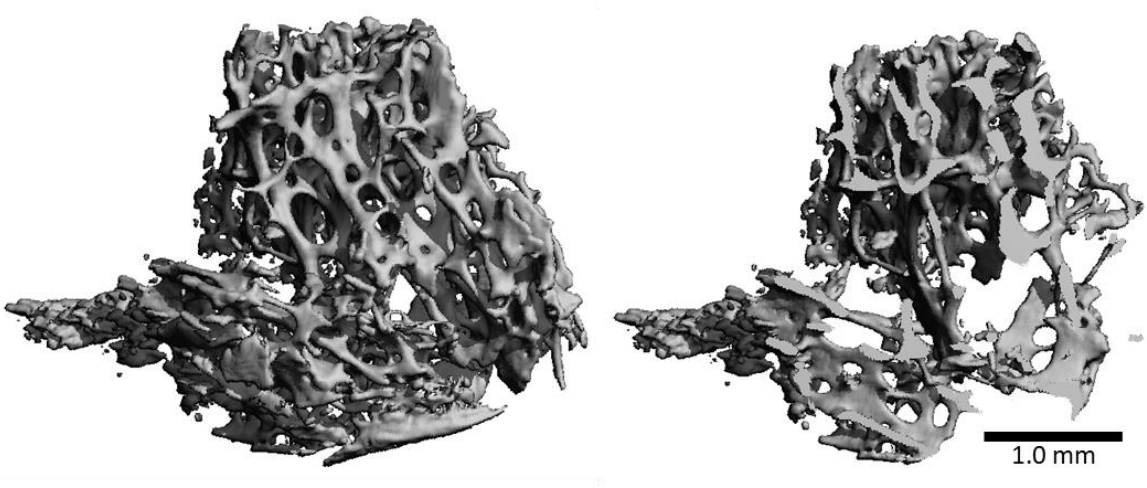


Figure 45 – 3mm rat tibial trabecular meshwork punch and micro-CT cross-section.

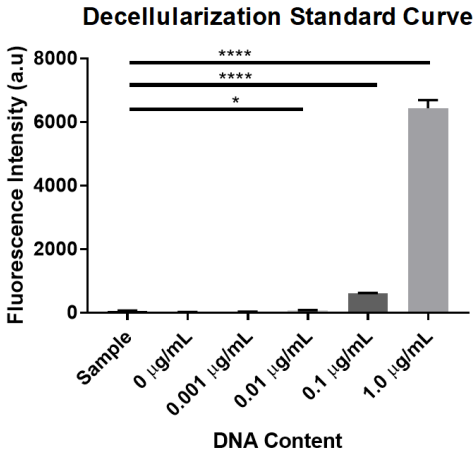


Figure 46 – Decellularization standard curve comparing DNA content of decellularized sample against known DNA content controls.

6.3.2.2 Scaffold loading

Prior to perfusion seeding, the decellularized meshwork or collagen sponge punches are loaded into a custom seeding chamber. The seeding chamber is comprised of two Luer-

Lock barbed adapters, a strip of Tygon tubing, and punched cell strainer mesh of 40 to 100 µm pore size with diameter equal to the inner diameter of Tygon tubing. The Tygon tubing is slipped over the first barb and the first punched cell strainer mesh is loaded into the chamber. We then load the scaffold into the tubing and place the second mesh on top of the scaffold. The second barb is then used to seal the opposite end and the barbs are permanently bonded to the tubing with epoxy to prevent air leaks. The system, shown in Figure 47, is then plugged in series with the fluidic circuit to run perfusion experiments and is split down the side with a razor to gain access to the scaffold post processing.

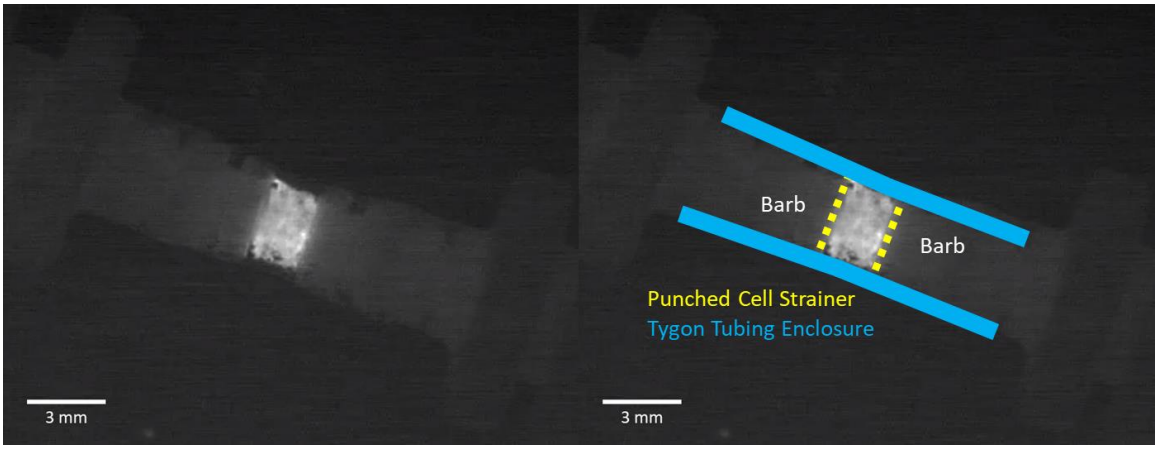


Figure 47 – Custom chamber with fluorescently tagged scaffold for visualization. Chamber is composed of two Luer-lock barbs placed inside Tygon tubing and biopsy-punched 100 µm pore size cell strainer placed over barb ends to hold scaffold in place.

6.3.2.3 Cell Culture and Staining

The lymphatic smooth muscle cell line, fluorescent labelling, and tracking technique, outlined by Hooks et al., was used for these experiments[117]. After culturing,

approximately 1.5 million cells were stained with 20 µl of 1 mg/mL LICOR 800CW NHS ester in DMSO for 20 minutes. Reaction was quenched with PBS and cells were washed with 10 mL of PBS twice. Cells were fixed and resuspended in 1 mL of PBS before infusion resulting a suspension of $1.5x10^6$ cell/mL.

6.3.2.4 Bead Experiment

The custom syringe system was retrofitted to behave as a perfusion bioreactor system[67]. Scaffolds were loaded into the perfusion bioreactor, and a suspension of 10 µm – 20 µm diameter microspheres at $6x10^5$ particles/mL were perfused through the decellularized scaffold at a superficial velocity of 3 mL/min for trabecular meshwork scaffolds. Micro-CT was used to determine seeding effects for silica microspheres in high porosity scaffolds.

6.3.2.5 Cell Seeding Experiment

Lymphatic smooth muscle cells modified with a fluorescent agent were used to seed collagen sponge to investigate density and uniformity. One mL of a suspension of $1.5x10^6$ cells/mL were perfused through collagen sponge at a rate of 3 mL/min. Fluorescent microscopy was used to record video data of scaffold seeding in the near infrared range for the duration of perfusion time. Both experiments were run for a $\phi = 1.00$ and 0.55. with a gross volume exchange of 0.0933 mL.

6.3.3 Results

Initial experiments were run to determine feasibility of scaffold seeding. These experiments took decellularized trabecular meshwork and perfused them with large silica microspheres with 15 μm average diameter. Silica microspheres were used to be able to easily differentiate the metal beads from the bone using micro-CT, Figure 48. This was done to determine functionality of the system and to optimize settings for cell seeding experiments.

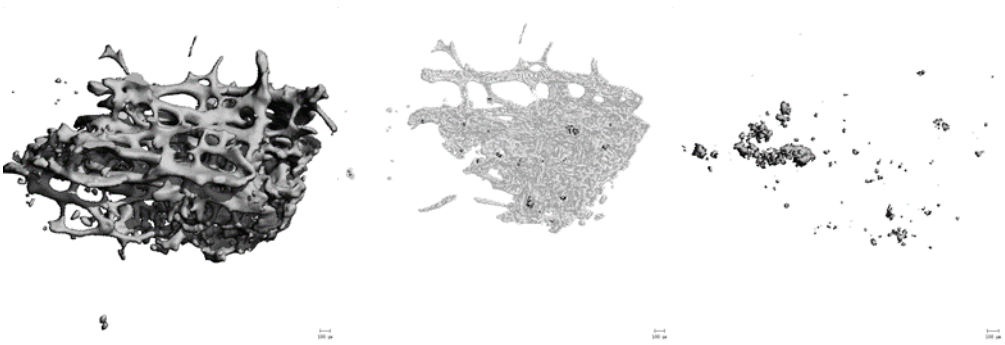


Figure 48 – Left to right: 3-D Micro-CT of murine tibial trabecular scaffold incubated with microspheres for 1 hour. (A) Scaffold and microspheres original image, (B) scaffold cross-section with microspheres highlighted, (C) Microsphere distribution through same scaffold isolated using imaging software.

Seeding experiments were run for static seeding and PM seeding of collagen sponge scaffolds. High resolution fluorescent microscopy video was captured, and line-by-line image intensity was measured for still images of $\phi = 1.00$ and $\phi = 0.55$ captured at the same time point. Figure 48, shows the line-by-line cross-sectional intensity of a seeded collagen sponge scaffold for $\phi = 1.00$ and $\phi = 0.55$. Figure 48 also shows the intensity of the image at each pixel mapped to a surface. We show that $\phi = 0.55$ has significantly higher

uniformity of seeding, Figure 50, observable in both the line and surface plots and calculated by Equation (5). Further, we can use these images to show behaviour of scaffolds like dead-end filtration with high intensity seeding near the surface of the scaffold which tappers to much lower levels for the $\phi = 1.00$ case, and improved perfusion of cells into the scaffold for $\phi = 0.55$.

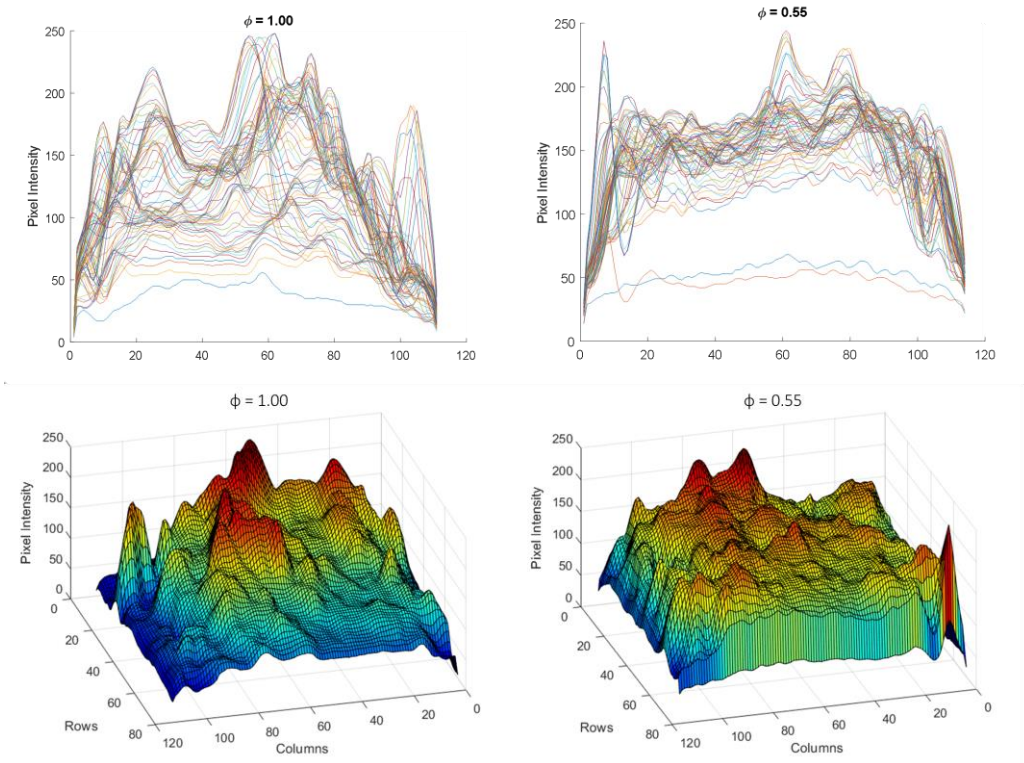


Figure 49 – Line-by-line intensity and surface plots for $\phi = 1.00$ and $\phi = 0.55$

A row-by-row percent uniformity, Equation (5), is calculated with the script defined in Appendix A.7[72]. Figure 50 plots the percent uniformity for $\phi = 1.00$ and $\phi = 0.55$. A significantly ($p<0.0001$) higher percent uniformity is observed for $\phi = 0.55$ with a median

value of 76.5% compared to ϕ = 1.00 at 67.7%. Further, we show a significant improvement in average intensity for the scaffold at ϕ = 1.00 of 112.8 and 143.4 for ϕ = 0.55.

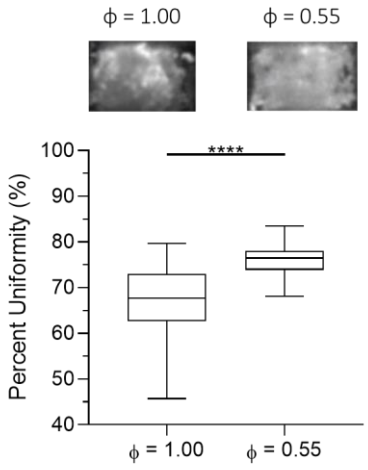


Figure 50 – Percent uniformity was calculated for each row from seeding images based, for three trials, on row intensity. Duty cycle ϕ = 1.00 and ϕ = 0.55 including scaffold voids are compared side by side with all row values of percent uniformity included herein. Representative images provided above grouping to demonstrate intensity variations.

The row-by-row percent uniformity calculation on whole images can incorporate voids along the periphery of the chamber housing. These voids will not contain cells and will pull the average intensity down and the standard deviation up, artificially biasing the observed results. In order to eliminate bias from the system, we set intensity values for photo pixels not containing scaffold equal to zero manually. We then modify the analysis script to exclude these data points from the calculation and rerun the calculations above.

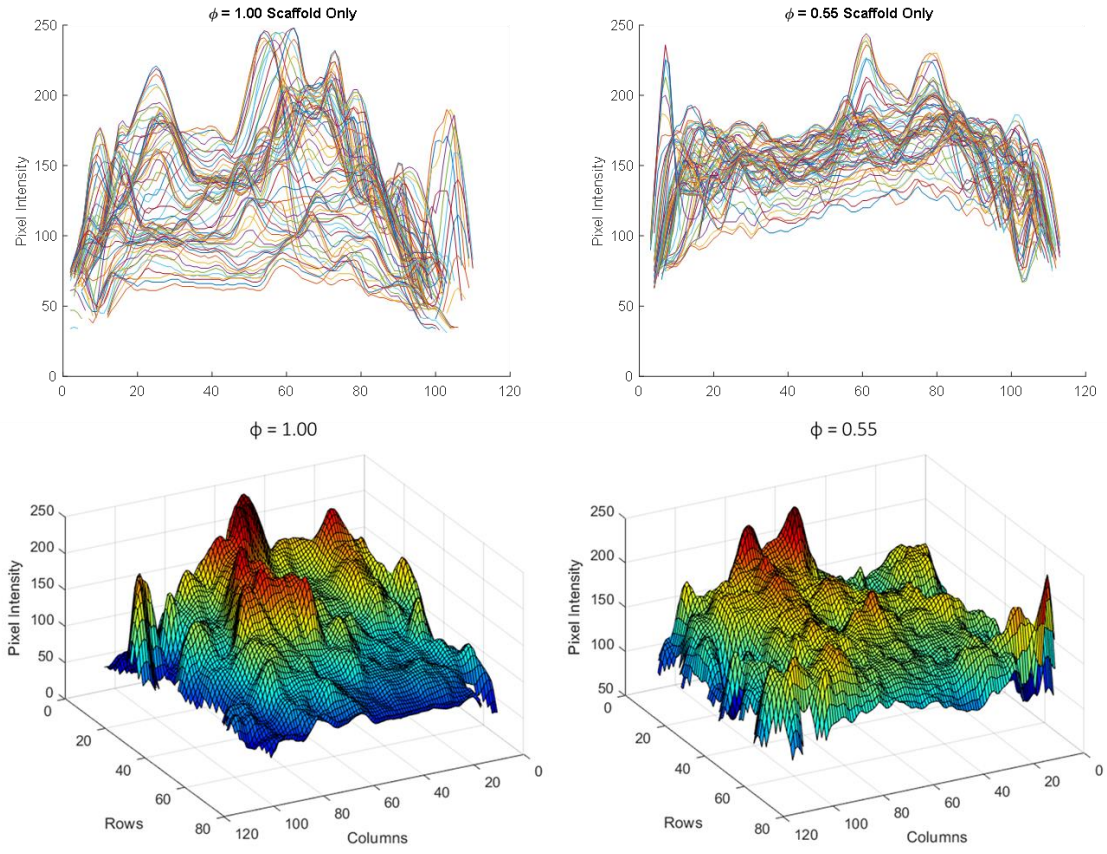


Figure 51 – Line-by-line intensity and surface plots excluding image sections not containing scaffold.

Figure 51 replots the intensity lines for each row of the image and surface plots to provide a perspective of an equivalent of surface roughness. Figure 52 plots the percent uniformity recalculated for ϕ = 1.00 and ϕ = 0.55 of the void-removed images. A significantly ($p<0.0001$) higher percent uniformity is observed for ϕ = 0.55 with a median value of 85.4% compared to ϕ = 1.00 at 78.2%. Further, we show a significant improvement in average intensity for the scaffold with ϕ = 1.00 producing an average intensity of 123.48 and 153.14 for ϕ = 0.55. The intensity average suggests a higher seed density in addition

to uniformity for ϕ = 0.55. The seed density can be calculated in the future by using a standard curve to relate the density of a known cellular seeding to an intensity value in the fluorescent microscopy image. The increase in average scaffold intensity confirms the pull-down effect that peripheral voids have on the overall image. Further, when investigating the percent uniformity for depth, we can take the vector of percent uniformity and look at mean uniformity in depth. We show a significant improvement in uniformity for depth of penetration with values of 95.4% and 89.2% for ϕ = 0.55 and ϕ = 1.00 respectively.

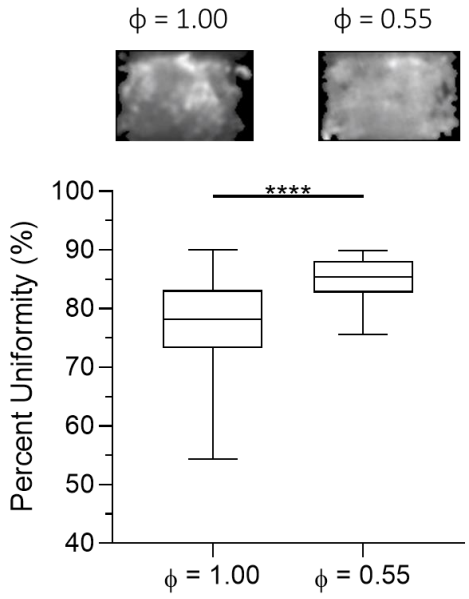


Figure 52 – Images for ϕ = 1.00 and ϕ = 0.55 and corresponding percent uniformity distributions for images with scaffold voids removed from the image n=3.

Finally, we compare the results of these two methodologies in a grouped system and against the current literature, Figure 53. As noted in Figure 50 and Figure 52 there are

significant differences between ϕ = 1.00 and ϕ = 0.55 for each independent approach. However, we show a significant difference between the two methodologies within the independent duty cycles. Further, when compared against the current literature, we show PM flow control at a duty cycle of 0.55 significantly outperforms oscillatory flow reported by Alvarez-Barreto et al[72] for both fiber and foam scaffolds.

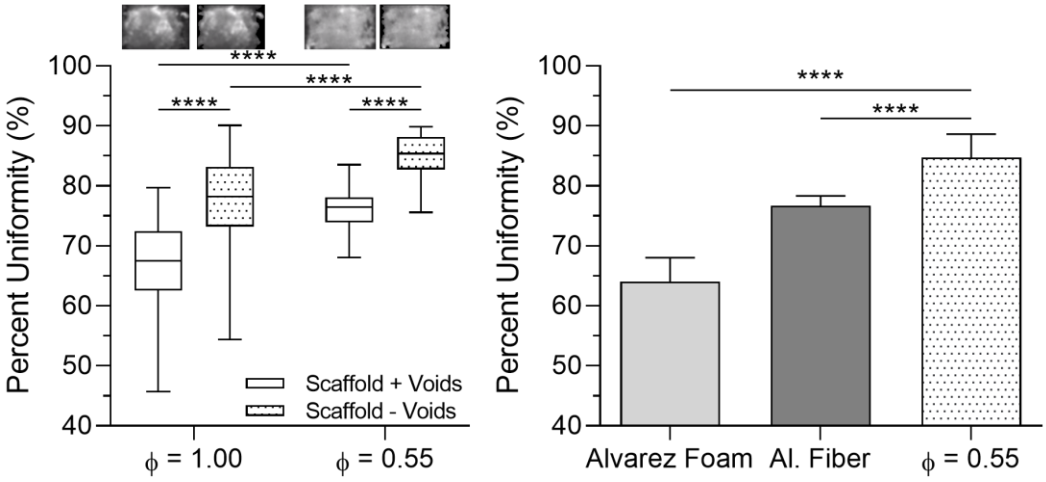


Figure 53 – Grouped comparison of percent uniformity for images with voids removed compared against whole images, n=3 for each group. Right image shows void removed uniformity compared against results achieved by Alvarez-Barreto et al. for both foam and fiber-based scaffolds when seeding collagen sponge.

6.3.4 Discussion

We investigated scaffold seeding utilizing PM backflush to improve perfusion of cells. We show significant improvement in both uniformity and seed density for scaffolds seeded utilizing PM periodic backflush with a median total percent increase of 8.8% or 7.2% when implementing periodic backflush and when eliminating peripheral voids.

Further, we show significant improvement in fluorescent intensity across the membrane indicating higher seed densities. The data supports the assumption that scaffolds exhibit behaviors akin to dead-end filtration, and as a result, the use of pulse modulated backflush could theoretically be used to tune scaffold penetration and cellular distribution. Decreasing fouling effects can also be observed in improved uniformity for depth of penetration through the membrane. However, uniformity and density are not all-encompassing metrics to use for identifying successful scaffold seeding. Experiments measuring viability, planar density, and morphological effects should be conducted in the future to understand the effects of PM in scaffold seeding.

CHAPTER 7. CONCLUSION

Section 7.1 provides a synopsis of the main findings and contributions of each specific aim discussed herein. Section 7.2 discusses the limitations of the work conducted and provide solutions that can be pursued to overcome the problems. Section 7.3 concludes the thesis by providing new potential applications of this research and additional experiments that can be pursued as continuations of this research.

7.1 Main Findings and Contributions

The main contribution of this research is a mechanism and optimization strategy for enhancing recovery percentage of targeted permeate in dead-end filtration. We show that particle recovery increases as a result of cake formation interruption, reintegration, and subsequent initiation of forward flow. We developed a model to provide insight into the effects of varying backflush volume and duty cycle and enable parameter selection desired for future work. We then applied these theories to industry relevant problems to validate the mechanism's practicality and significance. The concepts and process implementation are innovative mechanisms to restore flux capacity of current dead-end systems.

7.1.1 Overview of Specific Aim 1

The objective of Specific Aim 1 was to investigate PWM theoretically and experimentally as a proof-of-concept for enabling the use of dead-end systems to isolate

target particles. We showed that PWM backflush cyclically restores flux capacity of dead-end filters through clearance and reintegration of fouling layers with changes in pressure around the membrane and the associated variation in particle detachment. PWM backflush was shown to produce up to 18-fold higher permeate concentration and recovery percentages with a median of 54.7% compared to a baseline median of 7.1% for constant forward flow. Finally, we regressed a model against this data and used the best fit curve to map expected system performance across variations in duty cycle and backflush volume to create a surface plot of potential recovery percentages. This work is significant because it expands the capabilities of dead-end systems to perform at or above what can be achieved by cross-flow systems with significantly larger throughputs.

7.1.2 Overview of Specific Aim 2

The objective of Specific Aim 2 was to optimize the throughput of PWM backflush through modulation of pulse amplitude and frequency to minimize the observed reductions to throughput in Specific Aim 1. We show that, for square wave-based volume control, detachment and reintegration of the fouling layer is minimally dependent on backflush velocity. The observations align with literature in cross-flow studies which suggest linear dependence on rate of change of volumetric flow rate leaving backflush volume as the main flux capacity effector. PAM resulted in a comparable recovery percentage with 27.6% improvement to throughput, measured as a recovery percentage over processing time. PFM alone improved throughput by 15.8% for the fastest frequency tested. We attributed this

elevated throughput to minimization of cake formation prior to backflush and leveraging the systems discretized approximations of duty cycle to decrease time delay compensators within the system. Combined we show that PFM and PAM can be used to increase throughput by 38.8% compared to the PWM experiments. This work is significant because it shows that the trade-offs to throughput by increasing recovery percentage can be decreased without loss of recovery percentage gains.

7.1.3 Overview of Specific Aim 3

The objective of Specific Aim 3 was to use the PM-controlled flow concept to better isolate conjugated microspheres and bacteria and improve perfusion bioreactor scaffold seeding uniformity. We showed that PM flow control can improve recovery percentage of microspheres varying in size, material, and conjugation states, filtered from irregular debris fields, by an average of 3-fold. We then showed the impact of PM flow control on bacterial separations in a cystic fibrosis model. We produced a median fold change of 2.1 producing 88.8% recovery of bacterial CFUs and demonstrate consistency across track-etched and patterned microsieve filters. Finally, we show PM flow control can be used to dynamically seed collagen sponge scaffolds, increasing percent uniformity significantly higher than reported by Alverez-Barreto et al., and improving seed density over static techniques. This work is significant because it shows that the discovery and optimization validated in Specific Aims 1 and 2 can be used in industry relevant fields to increase yield and improve cell and tissue engineering.

7.2 Limitations and Potential Solutions

7.2.1 Scope of Experimentation

All experiments in CHAPTER 4 and CHAPTER 5 were conducted with a binary solution of microspheres with an ideal separation barrier between the two particle sizes. Suspensions were formed at a high concentration but binding and aggregation were limited with the addition of Tween20. We show that frequent interruption of the fouling layer at an optimal backflush volume optimizes recovery percentage for these suspensions. However, the response of this experimentation would likely change with variations in a few variables. Here we will discuss the anticipated effects related to these variables.

7.2.1.1 Size, Distribution, and Concentration

All experiments for PWM, PAM, and PFM were conducted with 7.32 and 2.19 μm diameter particles in a 50:50 ratio at a concentration of 5 million total particles per milliliter. We show with that smaller particles, down to 0.96 μm in diameter, can be successfully separated using these methods, and that particles as large as 15 μm in diameter can be easily removed. When considering separations of particles from other particles the critical parameter for consideration is the distribution of size in relation to the pore in the filter. For example, if the particle distributions overlap, e.g. 4 ± 1 μm and 5 ± 1 μm diameter, any stiff particles lower than the cutoff of the filter will permeate the membrane.

We show this in 6.1.3. These experiments showed that debris below a certain threshold permeated in greater quantities in addition to targeted particulate, maintaining the purity and enrichment ratios of the original sample. Further, the conjugated microsphere debris field showed that uniform distributions of larger particulate have only a minimal impact on fouling interruption using this method. The behavior shown in this study leads us to hypothesize that the main drivers affecting cake interruption are packing density and adhesion or irreversible binding. Any factor that may affect detachment of particulate will have negative effects on cake clearance resulting greater backflush volumes to produce membrane clearance.

The concentration and distribution of the debris field will also affect aggregate formation within the caking layer. Aggregation of particles can increase embedding, blocking, and other fouling types that will prevent permeate particulate flux. To mitigate the negative effects of aggregates, a surfactant solution can be used reduce inter-particle adhesion. Assuming a surfactant cannot be used, optimizing for smaller forward flow periods will minimize cake layer formation and reduce the time frame in which particles will interact with the membrane surface and other particles within the cake. Rapid and repeated backflush at low volumes with low duty cycle should abate general adhesion and associated reductions in recovery percentage gains through the use of PWM, PAM, and PFM.

7.2.1.2 Particle Deformability

The particles used for all experiments were either silica or polystyrene. Both of these materials are very stiff in relation to other materials or biological objects like cells. We hypothesize that deformable particles have a higher likelihood of embedding within the membrane. Particles embedded in the membrane may present hysteresis in the association and dissociation of the particles, potentially requiring higher negative transmembrane pressures to dislodge particles that were easily embedded. For particles that exhibit high viscoelasticity, PAM can be used to drive these larger requirements in TMP. However, for cellular products, this will need to be balanced against their individual tolerances for shearing, as larger TMP on embedded cells may increase shearing of cells and loss of targeted material or increases in contaminating agents. If PAM cannot be used to clear embedded viscoelastic particulate, this will likely increase irreversible deep bed fouling and blocking. To minimize these losses, you may be able to use select thinner membranes with smaller pore sizes to prevent embedding or deep bed fouling. Filter material can also be selected to prevent embedding or limit the interaction between the membrane surface and the particulate reducing the hysteresis in the system. Finally, slower forward flow rates creating smaller positive TMP may prevent embedding in the first place.

7.2.1.3 Adhesion

Adhesion is a major component affecting irreversible fouling and aggregate formation. Nonspecific binding was shown in 6.1 and 6.2 to have a negative effect on recovery percentage fold change compared against $\phi = 1.00$. This decrease in recovery

percentage fold change is compounded in more complex biological systems shown by the decrease in fold change from 3 to 2 in conjugated microsphere experiments compared against bacterial isolation from epithelial cells. This decline in fold change even though particle size is approximately equal is attributable to biofouling in bacterial suspensions. The addition of active binding components and protein production in combination with extracellular products results in higher aggregate formation and binding to surfaces which come into contact with the particles. Here, you can use similar approaches discussed in 7.2.1.1 to minimize total contact time and limit inter-particle and particle-membrane adhesion. However, suspended particle interactions may be able to be averted as well by selecting a duty cycle and backflush volume combination that maximizes convective currents within the backflush phases. Convective current strength and particle agitation will help to reduce contact and break up aggregated that may have formed within the cake.

7.2.1.4 Scale

All experiments conducted within the scope of this thesis were performed for small fixed volume amounts ranging from microfluidic systems to millifluidic systems with volumes processed in the range of 0.3 to 10 mL. At very small scales, prior literature has discussed that caking displacement may be a function of resonant frequency of the fluid within the channel[25]. Further, capacitive effects of the surrounding actuating equipment will have a larger effect on the motility of the fluid and the particles suspended therein. We compensated for these capacitive effects with the time delay compensator δ. As these

processing systems scale to process liters at a time, fluidic inertia becomes a more dominant effector within the circuit. Changing fluid direction will take more power and will result in higher TMP potentially leading to slower rates of change for volumetric flow rate and potentially membrane failure. Square wave behavior was selected for its rapid state change, and PAM experiments show that for near identical rates of change of flow rate, volume is the main driver in cake disruption. If the rate of change of flow rate is reduced significantly, detachment may require larger volumes resulting the need lower duty cycles with larger gross volume exchange per cycle through the membrane. Further, these large systems usually attempt to account for the larger volumes with larger membrane surface areas. This may serve as a boon for implementation of PWM, PAM, and PFM on these systems by increasing reverse TMP profiles and varied detachment across the membrane surface leading to greater reintegration of cake.

7.2.2 Hardware

7.2.2.1 Stepper Motor Actuation

The custom syringe actuator used in the experiments described in this thesis has digital commands converted to discrete steps for moving fluid. The controller's volumetric precision is directly tied to the stepper motor steps per revolution and volume of the actuating syringe. We experimentally determined average steps per mL for actuation of various syringes and show higher precision with smaller syringes with a trade off in maximum volumetric flow rate. To reduce the impact of this discretization and velocity

limitation, we recommend the use of smaller actuating syringe volumes, higher step per revolution motors, and a smaller pitch on the threaded rod used to convert rotational actuation to linear actuation.

7.2.2.2 Syringe Precision

Further, for the purposes of experimentation, we used BD disposable 3 mL syringes for the actuating syringe in all experiments. The syringes used have variations in inner diameter and soft rubber plungers for creating and maintaining pressures. These syringes contribute to capacitive effects in waveform implementation, decreasing volumetric flow rate changes. To overcome these limitations, we modified the 3D printed actuating component to enable the use of Hamilton Gastight High Precision Syringes with rigid stoppers. The redesign allows for the experimenter to affix the plunger of the syringe to the actuation plate with a screw, eliminating mounting-based plunger hysteresis.

7.2.3 Waveforms

The literature search conducted in this thesis showed membrane clearance reduction percentage is proportional to the rate of change of flow rate in cross-flow systems. Although detachment in cross-flow systems is primarily driven by shear forces on the membrane wall, we hypothesized large rates of change for flow rate will have similar effects in dead-end systems, leading to the selection of square waves for experimentation. This hypothesis was supported by the results of the PAM experiments when they showed

consistent recovery percentage for fixed backflush volume at varying backflush velocities. We also show in Chapter 4.4.3 that detachment dynamics play a role in reintegration of the fouling layer with the bulk concentrate. As designed, the system is currently unable to inject waves of a non-square profile, e.g. sine, triangle, sawtooth, or otherwise. These styles of wave packets would need to be independently programmed into the Arduino controller and LabView front end and referenced by the actuation script to run experiments on alternate waveforms. Further, these experiments would need to be conducted before exploring their integration into the refined model in Section 4.4.5 so that cake formation and clearance parameters could be properly regressed against experimental data.

7.3 Future Work

7.3.1 Ficoll-Free Leukocyte and Large Cell Sorting

Potential applications of label-free filtration include blood separation[63], pharmaceutical fractionation[118], and *in vivo* low-density lipoprotein separation from plasma[119]. PM periodic backflush has been shown to increase recovery percentage. As a result, we can also use PM periodic backflush to improve the purity of the retentate for large cell isolation and cell content fractionation in blood. To test these claims, we have already shown that we can improve the number of red blood cells that can be permeated through the membrane prior to fouling and have shown purity of 15 μm diameter microspheres in the retentate can be improved for blood diluted with DPBS. We spiked red blood cell samples with microbeads and compared PM modulated backflush of spiked

blood retentate to only forward flow. Future experiments can be performed to investigate live cell and leukocyte fractionation without the need for additives including Ficoll-paque.

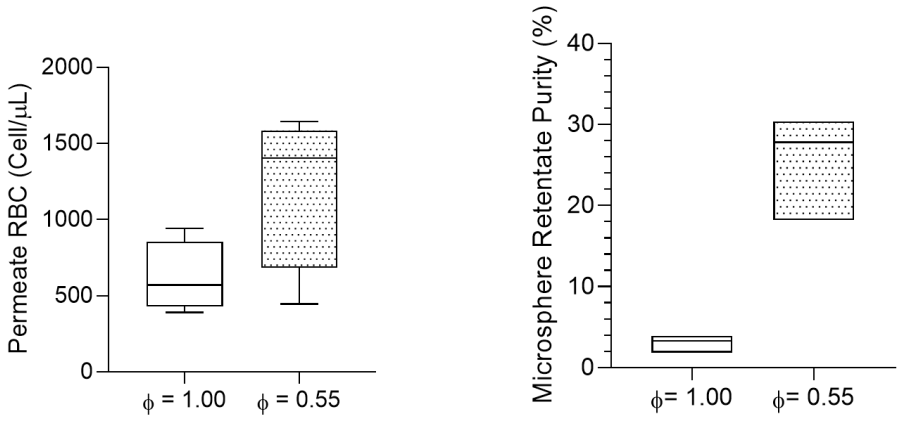


Figure 54 – Red blood cell concentration in permeate and resulting purity of microspheres in retentate for diluted blood using PM backflush processing.

Culture and spiking protocol for future work as follows:

Rare Cell Culture - Myelogenous leukemia, K562 cell line, will be cultured in media consisting of 89% RPMI base, 10% FBS, and 1% Penicillin-Streptomycin. Cells will be maintained in a CO2 incubator at 37 degrees Celsius and passaged every three to four days based on concentration. A hemocytometer will be used to establish concentration measurements from the suspension.

Spiking Protocol – Porcine blood, sourced from Dr. Ku lab, will be collected on the day of experimentation and appropriately diluted using Phosphate Buffered Saline solution as needed for experimental procedures. A small volume of myelogenous leukemia, K562,

cells at measured concentration will be added to the diluted porcine blood to achieve ratios of 1:100, 1:10,000, and 1:1,000,000 for rare cell compared to erythrocyte count.

7.3.2 Live Cell Scaffold Seeding

Preliminary results for scaffold seeding in Section 6.3.3 show that collagen sponge can be seeded with higher uniformity and cell density than static flow techniques. This is significant because it indicates that we can use PM backflush to reduce the impact of cellular fouling in static seeding techniques and increase penetration and uniformity for seeding larger scaffolds. There are two sets of experiments that can be pursued by expanding upon this foundational work.

Firstly, these experiments were run with dead, fixed smooth muscle cells. Investigating the effects of PM backflush on live cells to determine effectiveness of seeding and tissue formation will be critical to translating this work. Initial experiments are being performed with decellularized rabbit quadricep muscle and PLGA engineered scaffolds provided by Dr. Kent Leach of UC Davis using collagen sponge as a control. These experiments will pursue seeding of Mesenchymal Stem Cells, fibroblasts, and ovarian cancer cells. Initial results suggest that decellularized rabbit quadricep tissue is too compliant to withstand the pressure gradient produced inside the seeding chamber. The pressures compress the decellularized tissue producing a scaffold impermeable to cells, even at very low flow rates. To overcome this, we will be ensuring future punch biopsies are performed parallel to scaffold striations and may use a hypodermic needle to minimize

tissue collapse. However, preliminary results with collagen sponge as a scaffold show promising patterns of uniformity at a PM duty cycle of 0.55 compared to infusion with duty cycle equal to 1.00. Ultimately, we will need to investigate perfusion environmental conditions such as pH, temperature, and salinity in the bioreactor and test cell viability after seeding in tissue constructs.

Secondly, it would be significant to investigate the effects of PM periodic backflush on seeding very large scaffold structures. We show in Section 6.3.3 that we can overcome the dead-end filter like behavior of decellularized scaffolds when loaded into perfusion-based bioreactor system. Overcoming cellular fouling in large structures implies that we should be able to seed structures at greater depths and higher densities. Further, the refined model shows that we should be able to control desired recovery percentage output and, as a result, should be able to control seeding depth, uniformity, and density by precisely selecting duty cycle and backflush volume of our system. These experiments would enable the production of larger scaffold systems with the ability to seed different cell types in different patterns throughout a given scaffold.

7.3.3 Real-time Control of Backflush

Tracking performance and selecting desired outcomes given the results of the refined model is akin to finding a value in a lookup table; it is a useful approximation for most desired outcomes but is inefficient for complicated systems controlled by many variables. We hypothesize that this unnecessary burden can be eliminated in future work

by integrating real time pressure and concentration sensors to inform a PID controller which controls backflush duration and flow rate. Preliminary experiments integrated a differential pressure sensor to monitor transmembrane pressure. The controlling code was modified to capture TMP at 50 Hz intervals and display data to the monitor. Control software was also updated to handle control of inputs from the TMP sensor and use them to calculate proportional, integrative, and derivative components of the wave.

The results of the first TMP-based experiments showed that transmembrane pressure alone is insufficient to inform a controller at the scale on which we are operating. The system cannot use TMP as a controlling element because solvent flux is separate from the permeate flux and fouling has little impact on TMP. As a result, we incorporated permeate concentration data into the system as well. We used the data in Figure 55 to select a light source with a central frequency of 850 nm and built custom housing for an in-line photodetector and amplifier circuit to track absorption and relate it back to concentration at 50 Hz intervals.

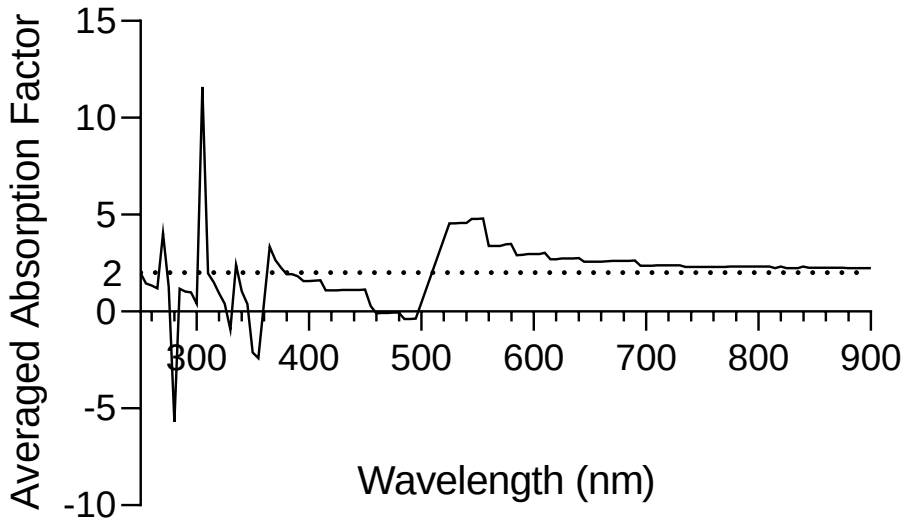


Figure 55 – Averaged wavelength absorption factor from iterative 2-fold serial dilution of polystyrene microsphere suspensions. Curve calculated from spectrophotometry data in Appendix B. Line at y equal to 2 shows ideal averaged absorption factor for creating a linear calibration curve.

We were able to demonstrate capture and output of absorption data but were unable to eliminate baseline shifts in current generated by the photodiode in our system. Future experiments will need to determine the cause of these baseline shifts and account for them before absorption data can be useful. Further, relating both absorption and TMP to desired recovery percentage outcomes and creating a controller informed by these variables will be non-trivial.

7.4 Conclusion

Pulse modulated periodic backflush is a useful and innovative approach to controlling fluid flow rate that contributes significantly to advancement and revitalization

of dead-end filtration systems. Using pulse modulated backflush, dead-end systems are able to outperform cross-flow filtration devices in both recovery percentage and throughput. We have shown that particle redistribution is a result of convective currents during backflush events and that, for square wave-based volume control, detachment and reintegration of the fouling layer is minimally dependent on backflush velocity. Additionally, we show that we can maximize system throughput by modulating the amplitude and frequency of flow rates. Finally, we demonstrate practical use cases in conjugated particle and cellular recovery and apply the technology to scaffold seeding to improve the uniformity and seed density significantly improving outcomes compared to what is currently available. This thesis serves to prove that pulse modulated backflush of dead-end filtration systems is the key to maximizing recovery percentage of targeted particles in fluidic suspensions and drives the restoration of flux capacity through clearance and reintegration of fouling layers.

APPENDIX A. SOFTWARE

Appendix A is a repository of all software generated over the course of this thesis. Each section of Appendix A will contain, as it appears in the main text, a title for the script in the first-level subheading, the software in which the script operates, a description of the output of the script, a description of how the script works in addition to in-line comments found in the structured code, and the structured code itself for the purposes of personal use through import or copy/paste.

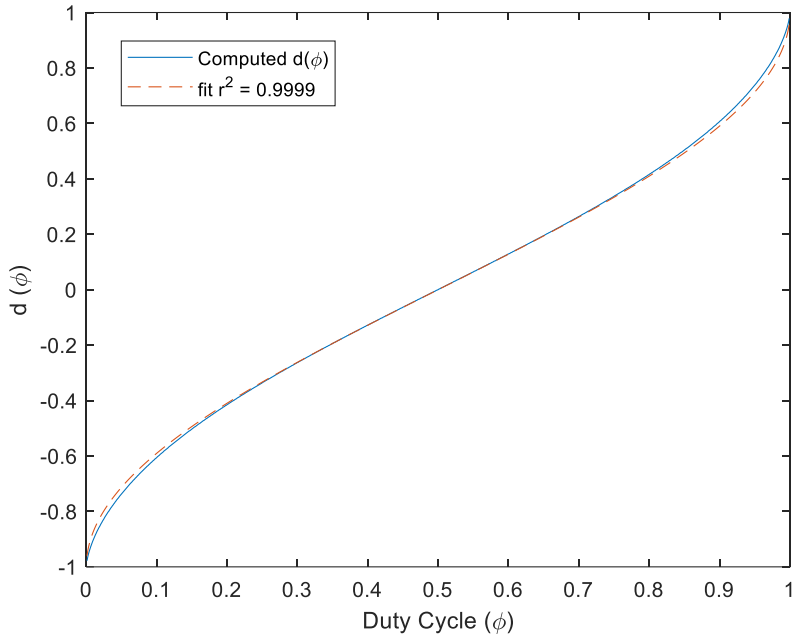
A.1 Quantitative Determination of d(ϕ)

A.1.1 Software

MATLAB

A.1.2 Output

This script uses computational calculations to find the values of d(ϕ) for a non-piecewise modulated sine wave. The script plots the calculated values of d and the best fit function derived using the built-in curve fitting tool. The base script was then modified to behave as a function which can be called by another to create a vector d(ϕ) that can be used for calculating $t_\alpha(\phi)$ without approximating the values through best fit.



A.1.3 Description of Function

This script is initialized by emptying values in the workspace, cleaning the command window, and closing all open figures. This is followed by the definition of constants for use in calculations including the period of the wave, the duty cycle range, and a vector of negative ones used to store the determined value of d. The script then utilizes anonymous functions to calculate the value of d at each duty cycle by incrementing the stored value and stopping only when the phi is greater than or equal to the expected value of phi. This value is stored for both the current value of d and the successive value of d in order to reduce computational time as the function d(ϕ) should be monotonically increasing. d(ϕ) is then plotted against duty cycle and the curve fitting tool is opened.

The modification of the script to behave as a function requires inputs of a vector for ϕ and the period of the wave. The script still plots the function and the best fit curve but will also return a vector for d(ϕ) to be used in downstream calculations.

A.1.4 Code

Script as an independent '.m' file

```
%% Reset workspace, clean command window, and close all figures
clear
clc
close all

%% Establish the constant variables
z=2*pi;                 %period of wave
phi=[0.001:.001:.999];      %Duty Cycle from 0:1
d=ones(1,length(phi)).*-1;  %Vector for calculations of d starting at -1

%% Generate anonymous functions to integrate over
f=@(t,c) (sin(t-asin(c))+c)./(1+abs(c));        %Q(t)
g=@(t,c) abs((sin(t-asin(c))+c)./(1+abs(c)));   %|Q(t)|

%% Calculate d for each phi by integrating the waves and checking phi
for i=1:length(phi)
   %% Calculate the Change in volume and the total volume change
   dV=integral(@(t)f(t,d(i)),0,z);
   Vtot=integral(@(t)g(t,d(i)),0,z);

   %% while the value of d produces a phi less than the anticipated
   while ((dV+Vtot)/(2*Vtot)<phi(i))
      d(i)=d(i)+.001;              %increase d by a nominal amount
      dV=integral(@(t)f(t,d(i)),0,z); %recalculate volume change
      Vtot=integral(@(t)g(t,d(i)),0,z);%recalculate volume total
   end

   %% To reduce time, start the next d at the current d
   if i<length(phi)
      d(i+1)=d(i);
```

```
    end
end

%% Plot the result and open the curve fitting tool.
plot(phi,d,phi,(2.*(asin((phi-.5).*2)./pi)),'--')
xlabel( 'Duty Cycle (\phi)')
ylabel ('d (\phi)')
legend('Computed d(\phi)','fit r^2 = 0.9999')
cftool % curve achieving best fit is a*(2*asin((x-.5)*2)/pi)
```

Script as an independent function to return d(ϕ)

```
function [d] = dOfPhi(phiVec,period)
%% Reset workspace, clean command window, and Close all figures
% clear
% clc
% close all

%% Establish the constant variables
z=period;                   %period of wave
phi=phiVec;      %Duty Cycle from 0:1
d=ones(1,length(phi)).*-1;  %Vector for calculations of d starting at -1

%% Generate anonymous functions to integrate over
f=@(t,c) (sin(t.*2.*pi./z-asin(c))+c)./(1+abs(c));       %Q(t)
g=@(t,c) abs((sin(t.*2.*pi./z-asin(c))+c)./(1+abs(c)));   %|Q(t)|

%% Calculate d for each phi by integrating the waves and checking phi
for i=1:length(phi)
    %% Calculate the Change in volume and the total volume change
    dV=integral(@(t)f(t,d(i)),0,z);
    Vtot=integral(@(t)g(t,d(i)),0,z);

    %% while the value of d produces a phi less than the anticipated
    while ((dV+Vtot)/(2*Vtot)<phi(i))
        d(i)=d(i)+.0001;                % increase d by a nominal amount
        dV=integral(@(t)f(t,d(i)),0,z);  % recalculate volume change
        Vtot=integral(@(t)g(t,d(i)),0,z);% recalculate volume total
    end

    %% To reduce time, start the next d at the current d
    if i<length(phi)
```

```
        d(i+1)=d(i);
    end
end

%% Plot the result and open the curve fitting tool.
plot(phi,d,phi,(2.*(asin((phi-.5).*2)./pi)),'--')
xlabel( 'Duty Cycle (\phi)')
ylabel ('d (\phi)')
legend('Computed d(\phi)','fit r^2 = 0.9999')
%cftool % curve achieving best fit is a*(2*asin((x-.5)*2)/pi)
```

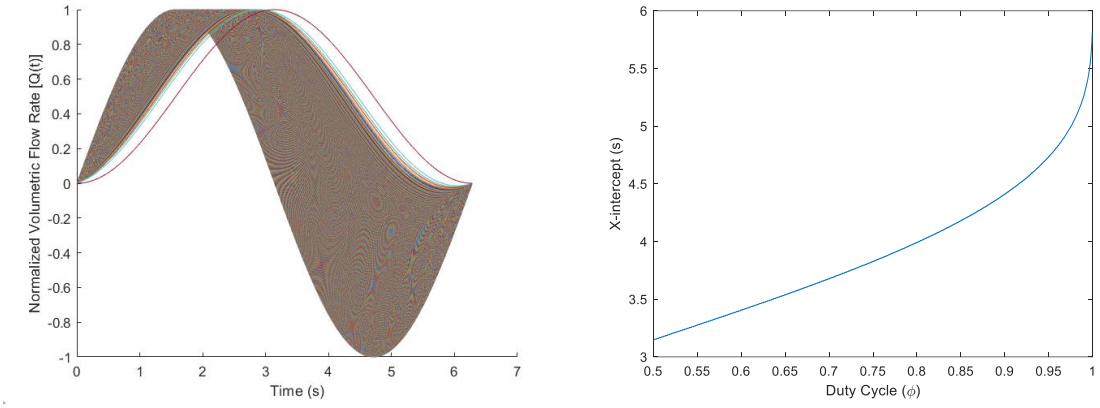
A.2 Quantitative Determination of $t_{\alpha}(\phi)$

A.2.1 Software

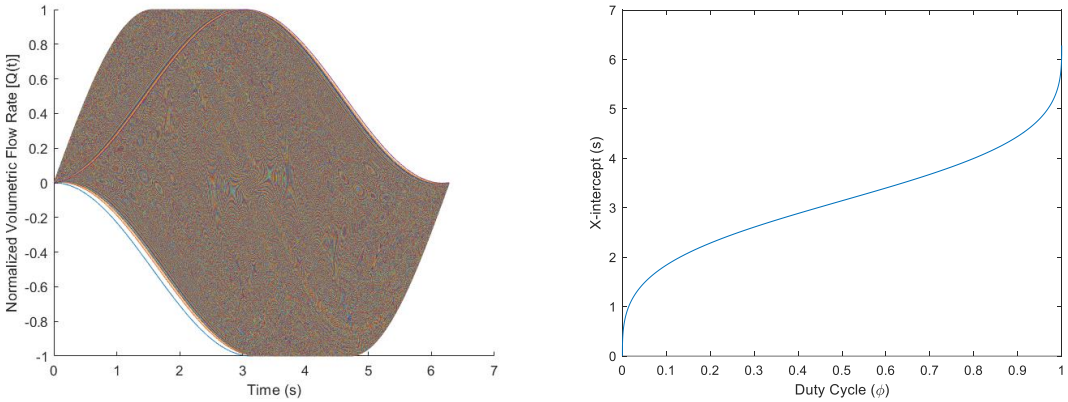
MATLAB

A.2.2 Output

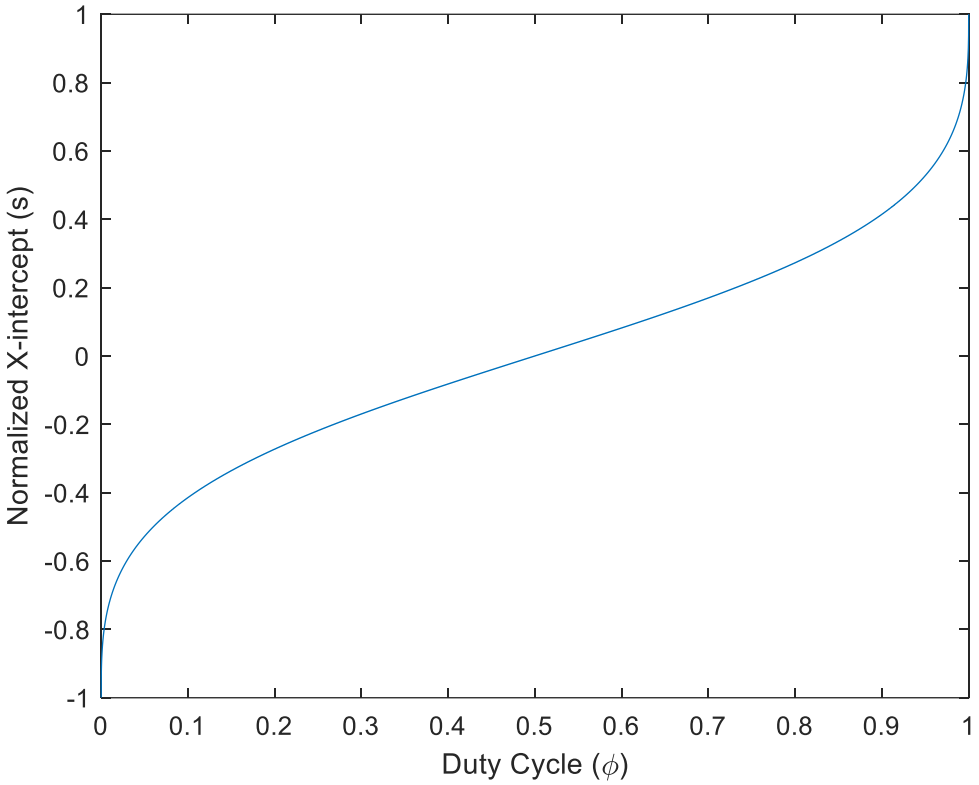
The script herein will output a normalized function Q(t) for each duty cycle, ϕ, ranging from 0.5 to 1.0. It will also output the x-intercept for the functions plotted as a function of ϕ and a normalized plot accounting for the period of the wave.



Plots of Q(t) for ϕ = 0.5 to 1 and corresponding $t_\alpha(\phi)$ not including ϕ = 1.00



Modified code to include ϕ from 0 to 1 and inclusive $t_\alpha(\phi)$ from 0 to 1



$t_\alpha(\phi)$ scaled normalized to the period of the wave for each value of ϕ

A.2.3 Description of Function

This script is initialized by removing values in the workspace, cleaning the command window, and closing all open figures. This is followed by the definition of constants for use in calculations including the duty cycle range, linearly spaced time function based on the length of the phi vector from zero to 2*π, the value of d for each duty cycle, and vectors of zeros used to store the determined value of $t_\alpha(\phi)$. The script then calculates $d(\phi)$ by calling the script in Appendix A.1and the values, Q(t), for each duty

cycle over the full period, plots the function against time for all duty cycles, crops the n*2π intercepts and determines the minimum of the absolute value of the cropped function. This t value is offset and stored in the intercept vector. The intercept vector is then plotted against duty cycle before being normalized by the wavelength and plotted against a normalized duty cycle.

A.2.4 Code

```
%% Reset workspace, clean command window, and Close all figures
clear
clc
close all

%% Establish the constant variables
phi=[0:.0001:1];                % Duty Cycle of the wave
z=2*pi;                         % period of wave
t=linspace(0,(2*pi()),length(phi)); % Time vector linearly spaced to phi
d=dOfPhi(phi,z);                % d(\phi) for all values of phi
intercept=zeros(1,length(phi));     % Intercept value vector set to 0's
intercept(length(intercept))=z;     % Last intercept value = period
g=zeros(1,length(phi));         % Predetermine length of cropped Q(t)
cropSize = 3;                   % size to crop the n*2pi intercepts

%% Calculate the x intercept, t_alpha, for each phi
figure
hold on
for i=1:(length(phi)-1)
    f=(sin(t.*2.*pi./z-asin(d(i)))+d(i))./(1+abs(d(i))); %define Q(t,phi)
    g = f(cropSize:(length(f)-cropSize));% Remove first and last intercepts
    dist = abs(g);              % make the intercept the minimum
    minDist = min(dist);        % determine the minimum value
    idx = find(dist == minDist);% find the index of the minimum in g
    plot(t,f)                   % plot each function f vs time
    intercept(1,i)=t(idx+cropSize);    % store the time of x-intercept
end
xlabel('Time (s)')
```

```
ylabel('Normalized Volumetric Flow Rate [Q(t)]')

%% Drop the final values and Plot the results
figure
plot(phi,intercept)                    % Plot the data
xlabel('Duty Cycle (\phi)')
ylabel('X-intercept (s)')

%% Normalize and shift the results to plot at t_alpha(0)=0
NormIntercept=(intercept./(z/2)-1);          % Scale and Shift intercepts
NormPhi=phi-.5;                        % Shift phi
figure
plot(phi,NormIntercept)                % Plot the values\
xlabel('Duty Cycle (\phi)')
ylabel('Normalized X-intercept (s)')
```

A.3 Arduino Control Software

A.3.1 Software

Arduino Wiring

A.3.2 Output

This script is designed to control a stepper motor actuator through a quadruple half H-bridge motor shield from a LabView interface. The script outputs a one of four digital outputs: (0,0) (1,0) (0,1) (1,1). These outputs control the direction of current flow and voltage levels through the stepper motor for a determined number of steps clockwise or anti-clockwise at provided rotational velocities. Further the program executes this process in loops to control the volume exchange as a function of duty cycle. While operating, the system checks to be certain that the zero indicator has not been hit and stops function if it

does and collects analogue to digital converted (ADC) data in 50 Hz increments for pressure and absorbance data. All collected data is saved in a comma delimited format, plotted in real time on a graph that can be exported to any software which can import comma delimited formats.

A.3.3 Description of Function

This script waits for a string input from a Serial Communication bus. The comma delimited string contains the duty cycle, gross volume exchange, syringe volume, mode of operation, flow rate of forward flow, flow rate of backflush, total volume for action, flags for collecting pressure and concentration data, and maximum delay between directional change as a compensator. The script parses this string into variables, scales the variables as appropriate for the syringe performing the actuation, then passes the variables into the main switch case structure. The switch case uses the mode determinant to execute the desired functionality through the motor controller. The Arduino executes in single step increments checking for contact with the zero point to prevent damage to the hardware in the reverse direction. While the selected process occurs, if the either data collection flags are true, the system collects and prints a comma delimited text string containing the ADC values to the Serial Communication bus. Upon completion, the code resets all variables and waits for a new input on the Serial Bus.

A.3.4 Code

```
#include "Stepper.h"
#include "math.h"

/*
 This version has built a scaffold of the actual feedback control system. The actual read of the photodiode has not been implemented. It should be done in line #119. Absorbance threshold is currently manually set, further work could be done about auto-determining the threshold based on the base line and a percentage.
 */
//Variable Declaration
int counter = 0;
String var1[15];
String test = "";
float var2[15];

int syringeVolume = 1; // used to determine which syringe is in the actuator
float volumeScalar = 0.00; //multiplier used to scale volumes entered based on actuating syringe

int stepByte = 0;   // for incoming serial data
float volume = 0;
unsigned short modeByte = 20;  // used to store the operational mode selected by the user
unsigned int oscNum = 0;     // stores the number of oscillations for the oscillation mode
const int stepsPerRev = 200;
int rpm = 100;      // rotations per minute
int backflushrpm = 100; //rotations per minute for backflush
float duty = 50.0;  // stores the fraction out of 100 dedicated to forward flow
const float backlash = 4.059; // average number of steps the stepper motor must take to initiate a change of direction
float alphaVolume = 0.0; // fraction of the total volume moved in the combined forward and backward phases dedicated to forward movement
float betaVolume = 0.0; // fraction of the total volume moved in the combined forward and backward phases dedicated to backward movement
float periodVolume = 0.0; // The total volume moved in the combined forward and backward phases
float remainingVolume = 0.0; // The remaining volume for the whole operation
int stepBackFlush = 0; // number of steps to complete a backflush volume exchange
int stepForwardFlush = 0;// number of steps to complete a forwardflush volume exchange
int actualStepForward = 0;
int maxDelay = 30000;
double dTa;
double dTb;
```

```
int dT;
bool = false;
int absorbBaseline = 0;
int pressureBaseline = 0;

//interrupt timer variables
unsigned short pressure = 0;  // used to store the currently measured TMP
unsigned short absorb = 0;  // used to store the currently measured absorbance
String pressureStr = "";
String absorbStr = "";
String dataString = "";
String com = ",";
String leadingZero = "0";

bool executionFlag = false; // used to start the timer interuption
bool recordPressureFlag = false;  // user input to determine if the user wants to record TMP data
//unsigned short pressureReading[16000];
bool recordAbsorbFlag = false;  // user input to determine if the user wants to record absorbance data
bool feedbackControlFlag = false; // user input to determine if the user wants to use feedback control according to absorbance
unsigned short feedbackThreshold = 0; // this is the threshold absorbance for feedback control
volatile bool stopStepperMotionFlag = false; // flag used to stop the stepper motion (only checked in PWM mode) on given condition
bool baselineFlag = false; // flag used to calculate baseline
volatile unsigned short baselineCount =0;  // used to count how many measurements are made in recording baseline and used to compute average
volatile unsigned int pressureSum = 0;  // used to sum over the PWM measurements, max = 65535 for unsigned int
volatile unsigned int absorbSum = 0;  // used to sum over the absorbance measurements, max = 65535 for unsigned int, potentially needs to be long

unsigned long timer;
unsigned long timer2;

// initialize the stepper library on pins 8 through 11:
Stepper myStepper = Stepper(stepsPerRev, 8, 9, 10, 11);

void setup()
{
```

```
  //set up the timer interrupts
  cli();//stop interrupts

 //set timer1 interrupt at 50Hz
  TCCR1A = 0;           // set entire TCCR1A register to 0
  TCCR1B = 0;   // same for TCCR1B
  TCNT1  = 0;   //initialize counter value to 0

  // set compare match register for 50hz increments
  OCR1A = 1249;  // = (16*10^6) / (50Hz*256) - 1 (must be <65536)

  // turn on CTC mode
  TCCR1B |= (1 << WGM12);

  // Set CS10 and CS12 bits for 256 prescaler
  TCCR1B |= (1 << CS12);

  // enable timer compare interrupt
  TIMSK1 |= (1 << OCIE1A);
  sei();//allow interrupts

  // open serial port, sets data rate to 9600 bps
  Serial.begin(115200);
  delay(15);

  while(Serial.available()>0){  //is there anything to read?
    char getData = Serial.read();  //if yes, read it
  }   // don't do anything with it.

  //set up ADC read pins as input
  pinMode(2, INPUT);
  pinMode(A3, INPUT);
  pinMode(A4, INPUT);
}

//timer1 interrupt 50Hz records pressure data from analog pins for pressure and absorbance
ISR(TIMER1_COMPA_vect) {
  if (executionFlag){
    //checks if pressure flag is high and records a value and prints if so
    if (recordPressureFlag) {
      pressure = analogRead(A3);
```

```
    //add leading zeroes to the string for reading on the labview display end to normalize
string length
    if(pressure < 10){
      pressureStr = String(leadingZero + leadingZero + leadingZero + pressure);
    }
    else if (pressure < 100){
      pressureStr = String(leadingZero + leadingZero + pressure);
    }
    else if (pressure < 1000){
      pressureStr = String(leadingZero + pressure);
    }

  }
  //checks if absorbance flag is high and then record, print and stop the stepper on threshold
  if (recordAbsorbFlag) {
    absorb = analogRead(A4);
    if(absorb < 10){
      absorbStr = String(leadingZero + leadingZero + leadingZero + absorb);
    }
    else if (absorb < 100){
      absorbStr = String(leadingZero + leadingZero + absorb);
    }
    else if (absorb < 1000){
      absorbStr = String(leadingZero + absorb);
    }
  }
  // only the sum is calculated here, average is calculated after baseline measurements
//   if (baselineFlag){
//     baselineCount++;
//     if (recordPressureFlag) {
//     pressureSum += pressure;
//     }
//     if (recordAbsorbFlag) {
//     absorbSum += absorb;
//     }
//   }
//   stopStepperMotionFlag = false;
//   if (feedbackControlFlag) {
//       if (absorb>feedbackThreshold) {
//         stopStepperMotionFlag = true;
//       }
//
```

```
  dataString = String(pressureStr + com + absorbStr + com);
  Serial.print(dataString);
  }
}

//Main loop for execution and control of the motor shield and stepper actuator
void loop() {
  //Wait for an input from labview
    while (Serial.available() == 0); {
    }
    //Store all values in a vector
    while (Serial.available()>0) {
    var1[counter] = Serial.readStringUntil(',');
    test = var1[counter];
    var2[counter] = test.toFloat();

    counter++;
    timer2=millis();
    while(millis()-timer2<100){}
  }
  Serial.read();

  timer2=millis();
  while(millis()-timer2<100){}

  //store the vector values into independent variables
  duty = var2[0];
  syringeVolume = var2[2];
  if (syringeVolume == 0) {
    volumeScalar = 0;
  }
  else if (syringeVolume == 1) {
    volumeScalar = 1;
  }
  else if(syringeVolume == 2){
    volumeScalar = 0.347823;
  }
  else {
    volumeScalar = 1.425;
  }
  periodVolume = var2[1]*volumeScalar;
  modeByte = int(round(var2[3]));
```

```
if (modeByte == 3 && syringeVolume == 3) {
  volumeScalar = 1.3765;
}
rpm = var2[4];
backflushrpm = int(var2[5]);
maxDelay = int(round(var2[6]*1000));
volume = float(var2[7]*volumeScalar);
recordPressureFlag = bool(round(var2[8]));
recordAbsorbFlag = bool(round(var2[9]));

counter=0;

//calculate variables based on inputs
backflushrpm = round(rpm*(volumeScalar));
rpm = round(rpm*volumeScalar);
stepBackFlush = round(volume * (200 / 0.04556) + backlash); //backlash considered here
stepForwardFlush = round(volume * (200 / 0.04556) + backlash);

//report back to LabView the number of steps for back and forward flow if PWM
if(modeByte == 1){
  dataString = String(stepBackFlush);
}
else if(modeByte == 2){
  dataString = String(stepForwardFlush);
}
Serial.print(dataString);
Serial.flush();

//execute action based on provided mode selection
switch (modeByte) {

  //zero the system
  case 0: {
      myStepper.setSpeed(rpm);
      //Step backcward until zero contact is made
      while (!digitalRead(2)) {
        myStepper.step(1);
      }
      break;
    }

  //infusion
```

```
    case 1: {
      if (recordPressureFlag||recordAbsorbFlag) {
        //initialize timer 2
        timer2 = millis();

        //enable computing baseline
        baselineCount = 0;
        pressureSum = 0;
        absorbSum = 0;
        baselineFlag = true;
        executionFlag = true;

        //wait 2 second to calibrate either sensor, cannot use delay, prevents interrupt
        while ((millis() - timer2) < 2000) {
        } // end while

        if(recordPressureFlag){
          pressureBaseline = pressureSum/baselineCount;
//          Serial.print("PressureBaseline: ");
//          Serial.println(pressureBaseline);
        }

        if(recordAbsorbFlag){
          absorbBaseline = absorbSum/baselineCount;
//          Serial.print("AbsorbanceBaseline: ");
//          Serial.println(absorbBaseline);
        }

        baselineFlag = false;
      }

      //step backward until system reaches designated steps or contact with zero
      myStepper.setSpeed(backflushrpm);
      for (int i = 0; i < stepBackFlush; i++) {
        if (!digitalRead(2)) {
          myStepper.step(1);
        }
      }
      delay(15);
      break;
    }
```

```
    //Withdrawal
    case 2: {
       if (recordPressureFlag||recordAbsorbFlag) {
        //initialize timer 2
        timer2 = millis();

        //enable computing baseline
        baselineCount = 0;
        pressureSum = 0;
        absorbSum = 0;
        baselineFlag = true;
        executionFlag = true;

        //wait 2 second to calibrate pressure sensor
        while ((millis() - timer2) < 2000) {
        } // end while
        if(recordPressureFlag){
          pressureBaseline = pressureSum/baselineCount;
//          Serial.print("PressureBaseline: ");
//          Serial.println(pressureBaseline);
        }
        if(recordAbsorbFlag){
          absorbBaseline = absorbSum/baselineCount;
//          Serial.print("AbsorbanceBaseline: ");
//          Serial.println(absorbBaseline);
        }
        baselineFlag = false;
       }

       myStepper.setSpeed(rpm);
       for (int i = 0; i < stepForwardFlush; i++) {
          myStepper.step(-1);
       }
       delay(15);
       //Serial.println(millis()-timer2);
       break;
      }

    //PWM withdrawl
    case 3: {
            // case of using feedback control
      if (feedbackControlFlag){
```

```
//       remainingVolume = volume;
//
//    //Calculate necessary variables
//    //alpha Volume
//       alphaVolume = periodVolume * duty / 100;
//
//       //Beta Volume
//       betaVolume = ((100 * alphaVolume) / duty) - alphaVolume;
//
//       stepForwardFlush = round(alphaVolume * (200 / 0.04556));
////         Serial.print("max step forwardflush is ");
////         Serial.println(stepForwardFlush);
//
//       stepBackFlush = round(betaVolume * (200 / 0.04556));
////         Serial.print("step backflush is ");
//  dataString = String(stepForwardFlush + com + stepBackFlush + com
+(stepForwardFlush-stepBackFlush));
//       Serial.println(dataString);
//
//       oscNum = abs(round(volume / (alphaVolume - betaVolume)));
//       if ((alphaVolume - betaVolume) == 0) {
//         Serial.print("Enter Number of Oscillations (int)  ");
//         while (Serial.available() == 0) {
//         }
//         oscNum = Serial.parseInt();
//         Serial.println(oscNum);
//       }
//
//       dTa = maxDelay / (1 + pow(2.71828, (0.25 * (duty - 75))));
//       dTb = maxDelay / (1 + pow(2.71828, (0.25 * ((100 - duty) - 75))));
//       dT = round(dTa * dTb / maxDelay);
//       //    Serial.println(dT);
//
//       if (recordPressureFlag||recordAbsorbFlag) {
//         //initialize timer 2
//         timer2 = millis();
//
//         //enable computing baseline
//         baselineFlag = true;
//         executionFlag = true;
//         baselineCount = 0;
//         pressureSum = 0;
```

```
//        absorbSum = 0;
//        //wait 2 second to calibrate pressure sensor
//        while ((millis() - timer2) < 2000) {
//        } // end while
//        if(recordPressureFlag){
//          pressureBaseline = pressureSum/baselineCount;
//          Serial.print("PressureBaseline: ");
//          Serial.println(pressureBaseline);
//        }
//        if(recordAbsorbFlag){
//          absorbBaseline = absorbSum/baselineCount;
//          Serial.print("AbsorbanceBaseline: ");
//          Serial.println(absorbBaseline);
//        }
//        baselineFlag = false;
//      }
//
//      timer2 = millis();
//
//      while (remainingVolume > 0) {
//        //Serial.print("RemainingVolume: ");
//        //Serial.println(remainingVolume);
//        stopStepperMotionFlag = false;
//        if (stepForwardFlush > 0) {
//          myStepper.setSpeed(rpm);
//          Serial.println("Forward");
//          for (int i = 0; i < (stepForwardFlush + backlash); i++) {
//            //if (!digitalRead(2)) {
//            // stop forward flow when threshold is reached and duty cycle > 50%
//            if(stopStepperMotionFlag&&(i>(stepBackFlush + backlash))){
//              break;
//                                    actualStepForward = i;
//            }
//            actualStepForward = i;
//            myStepper.step(-1);
//          } //end for
//                          remainingVolume -= (actualStepForward - backlash -
stepBackFlush) / (200 / 0.04556);
//            //} //end if
//          }//end if
//          //timer was here
//          //if (!digitalRead(2)) {
```

```
//          timer = millis();
//          while ((millis() - timer) < dT) {
//          } // end while
//          //}
//
//        if (stepBackFlush > 0) {
//          myStepper.setSpeed(backflushrpm);
//          Serial.println("Backward");
//          for (int i = 0; i < (stepBackFlush + backlash); i++) {
//            //if (!digitalRead(2)) {
//            myStepper.step(1);
//            //} //end if
//          }//end for
//          //timer was here
//          //if (!digitalRead(2)) {
//          timer = millis();
//          while ((millis() - timer) < dT) {
//          } // end while
//          //}
//        }//end for
//
//        /* if (!recordPressureFlag&&!recordAbsorbFlag) {
//          if (j == round(oscNum / 4)) {
//            Serial.println("25% Complete.");
//          }
//          else if (j == round(oscNum / 2)) {
//            Serial.println("50% Complete.");
//          }
//          else if (j == round(3 * oscNum / 4)) {
//            Serial.println("75% Complete.");
//          }
//          else if (j == (oscNum - 1)) {
//            Serial.println("100% Complete!");
//          }
//        } feedback control must have some sort of measurements*/
//      }
    } //end if


        // case of not using feedback control
    else{
      //Calculate necessary variables
```

```
      //alpha Volume
      alphaVolume = periodVolume * duty / 100;

      //Beta Volume
      betaVolume = ((100 * alphaVolume) / duty) - alphaVolume;

      stepForwardFlush = round(alphaVolume * (200 / 0.04556));
      stepBackFlush = round(betaVolume * (200 / 0.04556));

      dataString = String(stepForwardFlush + com + stepBackFlush + com +
(abs(stepForwardFlush-stepBackFlush)));
      Serial.print(dataString);

      Serial.flush();

      timer2=millis();
      while(millis()-timer2<100){}

      oscNum = abs(round(volume / (alphaVolume - betaVolume)));

      dTa = maxDelay / (1 + pow(2.71828, (0.25 * (duty - 75))));
      dTb = maxDelay / (1 + pow(2.71828, (0.25 * ((100 - duty) - 75))));
      dT = round(dTa * dTb / maxDelay);

      if (recordPressureFlag||recordAbsorbFlag) {
        //initialize timer 2
        timer2 = millis();

        //enable computing baseline
        baselineCount = 0;
        pressureSum = 0;
        absorbSum = 0;

        baselineFlag = true;
        executionFlag = true;

        //wait 2 second to calibrate pressure sensor
        while ((millis() - timer2) < 2000) {
        } // end while
        if(recordPressureFlag){
          pressureBaseline = pressureSum/baselineCount;
        }
```

```
        if(recordAbsorbFlag){
          absorbBaseline = absorbSum/baselineCount;
        }

        baselineFlag = false;
      }

      timer2 = millis();

      for (int j = 0; j < oscNum; j++) {

        if (stepForwardFlush > 0) {
          myStepper.setSpeed(rpm);
//          Serial.println("Forward");
          for (int i = 0; i < (stepForwardFlush + backlash); i++) {
            myStepper.step(-1);
          }

          timer = millis();
          while ((millis() - timer) < dT) {
          } // end while
          //}
        }//end if

        if (stepBackFlush > 0) {
          myStepper.setSpeed(backflushrpm);
//          Serial.println("Backward");

          for (int i = 0; i < (stepBackFlush + backlash); i++) {
            myStepper.step(1);
          }//end for

          //timer was here

          timer = millis();
          while ((millis() - timer) < dT) {}
        }  //end if
      }

    } //end else
    break;
```

```
}

//Oscillate
case 4: {
    for (int j = 0; j < (oscNum); j++) {
      myStepper.setSpeed(backflushrpm);
      for (int i = 0; i < (stepBackFlush); i++) {
        if (!digitalRead(2)) {
          myStepper.step(1);
        }
      }
      timer = millis();
      while ((millis() - timer) < 150) {
      }

      myStepper.setSpeed(rpm);
      for (int i = 0; i < (stepBackFlush); i++) {
        if (!digitalRead(2)) {
          myStepper.step(-1);
        }
      }
      timer = millis();
      while ((millis() - timer) < 150) {
      }
    }
  }

//Priming
case 5: {
    //Serial.println("Enter 0 to zero system.");
    while (Serial.available() == 0) {
    }
    modeByte = Serial.parseInt();
    myStepper.setSpeed(100);
    while (!digitalRead(2)) {
      myStepper.step(1);
    }

   // Serial.println("Enter 0 to withdraw 0.6 mL.");
    while (Serial.available() == 0) {
    }
    modeByte = Serial.parseInt();
```

```
      stepForwardFlush = round(0.6 * volumeScalar * (200 / 0.04556) + backlash);
      for (int i = 0; i < stepForwardFlush; i++) {
        myStepper.step(-1);
      }

      //Serial.println("Enter 0 to zero system.");
      while (Serial.available() == 0) {
      }
      modeByte = Serial.parseInt();
      while (!digitalRead(2)) {
        myStepper.step(1);
      }

      break;
    }


  } //end switch

  //reinitialize all variables or set them outside acceptable ranges to be changed
  executionFlag = false;
  recordPressureFlag = false;
  recordAbsorbFlag = false;
  baselineFlag = false;
  modeByte = 20;
  periodVolume = 0.0;
  feedbackControlFlag = false;

  timer2=millis();
  while(millis()-timer2<100){}

  //Send flag to LabView that recording is done
  Serial.print("endthisrun");
}
```

A.4 Front-end GUI

A.4.1 Software

LabView

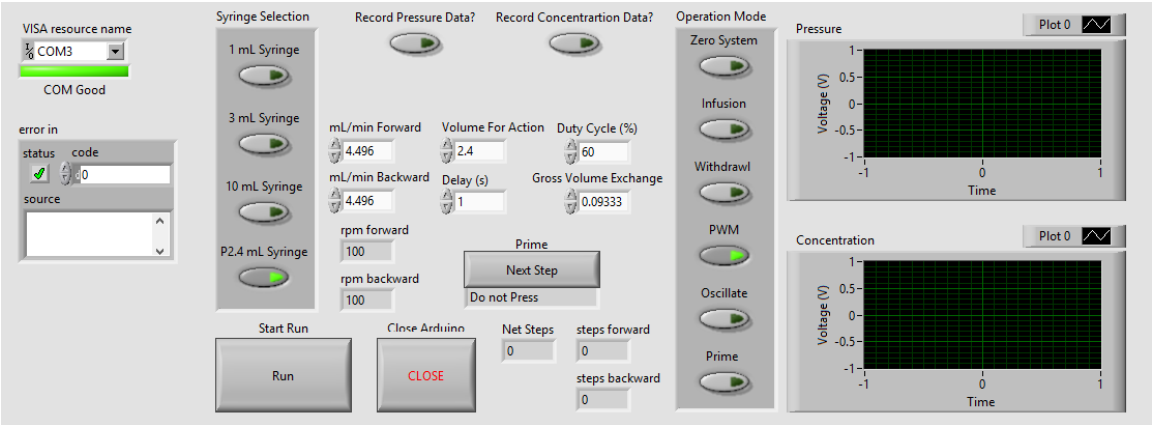
A.4.2 Output

This script is designed to take inputs entered on the front end and communicate them to a microcontroller through a comma delimited string over a Serial Communication bus. This script also takes inputs from a microcontroller in the form of comma delimited strings, parses them, and displays them in real time on an adjusting scale.

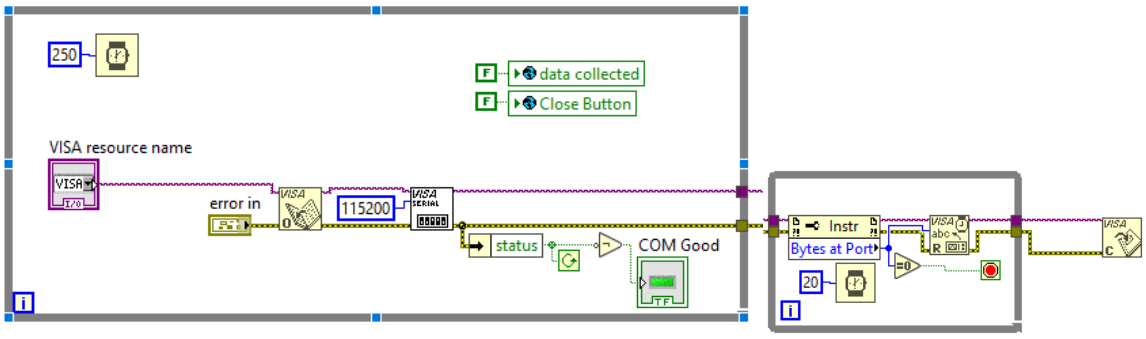
A.4.3 Description of Function

The GUI provides several pull-down menus, radio buttons, text fields, push buttons, toggle switches, and graphical outputs. After connecting over USB-A to USB-B to an Arduino microcontroller, the user selects the Serial Communication VISA from a self-updating pull down menu. Once connected, the user selects the syringe used for actuation, the operation mode, whether to record pressure or concentration data, and the values to send to the microcontroller to set the variables for execution. Upon selecting ‘Run’ these values are converted from numbers, binaries, and radio button selections to a comma delimited string and transmitted over Serial Communication to a microcontroller. Depending on the selections made, the system may execute loops in predetermined fashions to facilitate ease of experimental setup and execution, simulating autonomy. Further the code receives comma delimited strings and parses these strings into data that is stored in a vector and plotted in real time on self-adjusting graphs. Upon completion of the prior execution, the script waits for new input or termination of the program by the user.

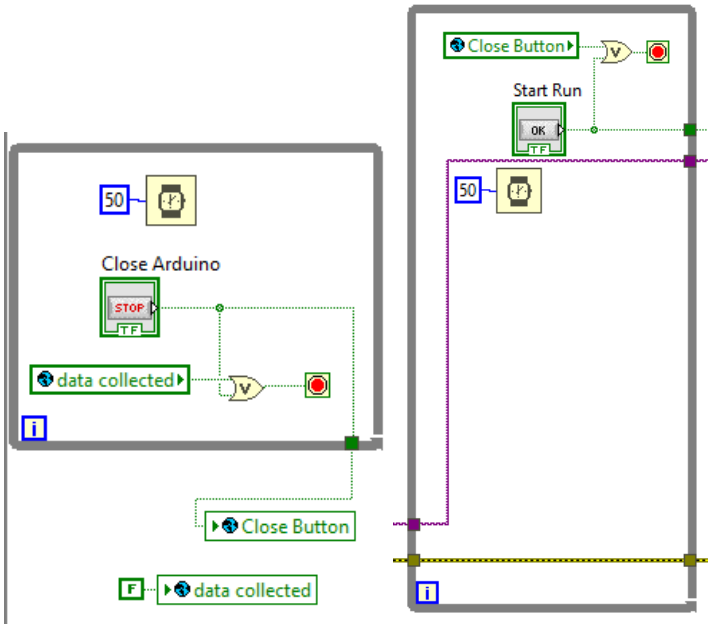
A.4.4 Code



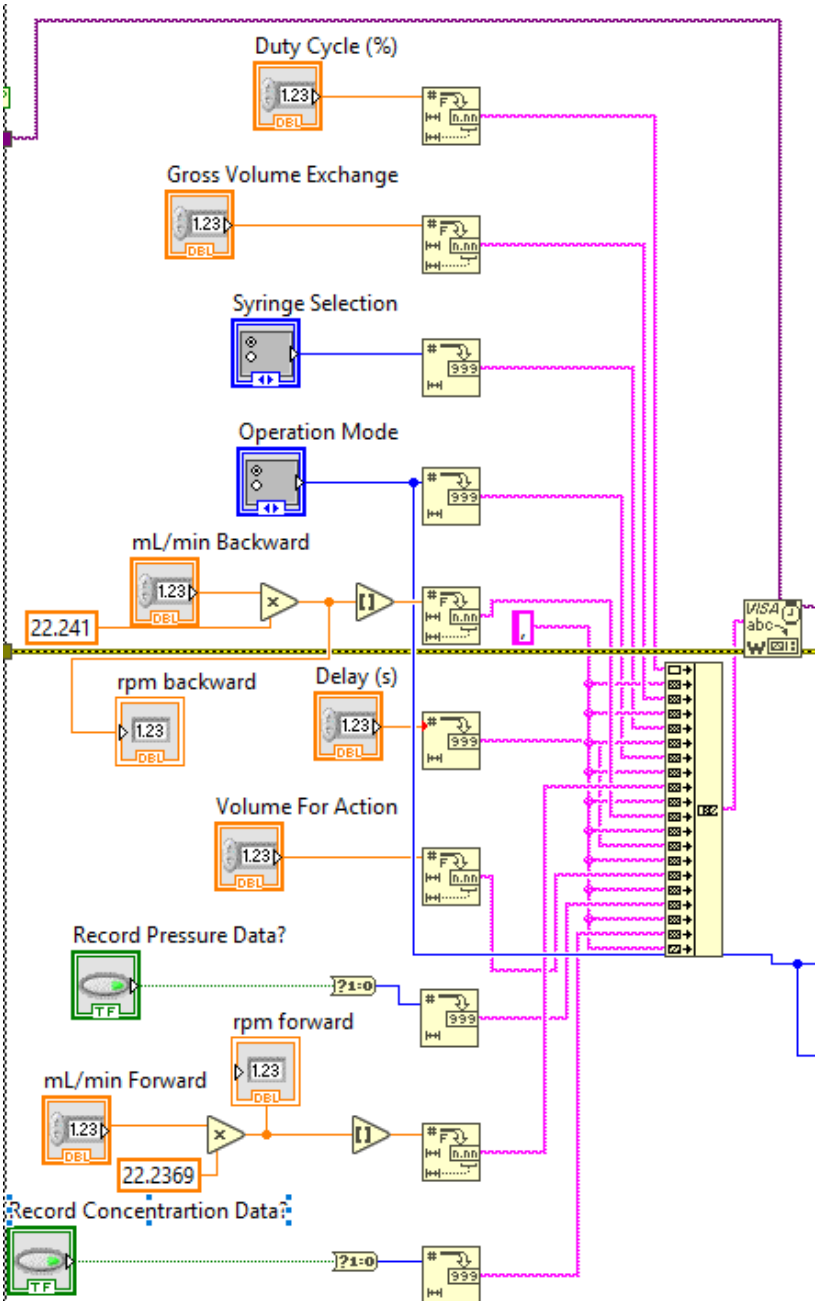
User facing application, saved and run as a '.exe' file. Allows for interaction by the user to simulate button pushing and provide text fields for inputs.



When run, the system declares two global variables controlling exit and data collection flags while checking for a VISA resource connection, selected by user. Once selected, the system establishes a connection with the Arduino and outputs a connection status indicator. Upon termination of the program, the system flushes the serial port, and closes the VISA resource.

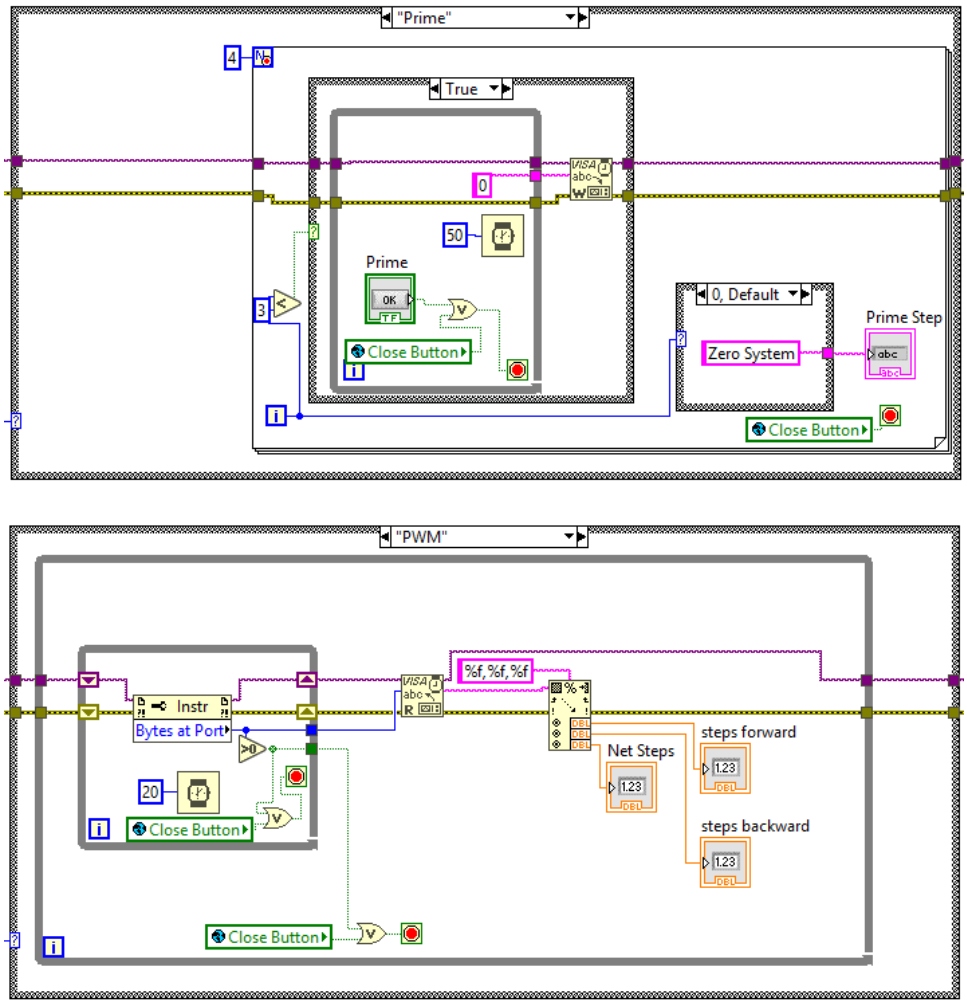


Once the connection to the VISA resource is established, the system waits for the user to make the desired changes and waits for an input of either a start button toggle or a close button toggle. These modify the global variables and loop until a state change is detected.



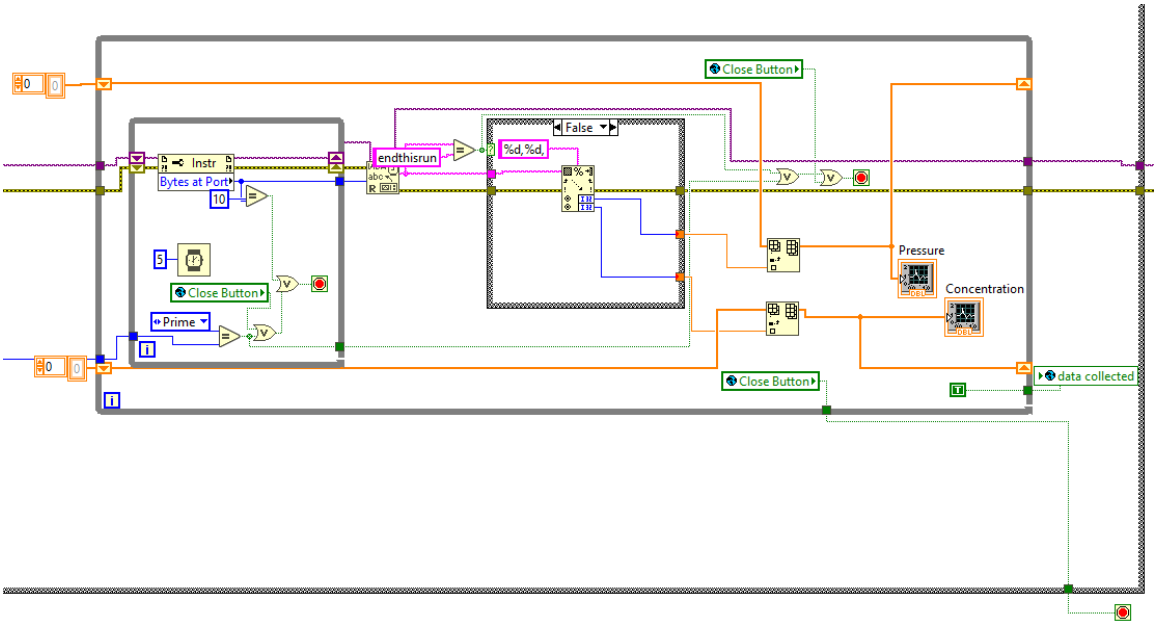
After the user depresses the ‘Run’ button, the system takes all the selections on the GUI front panel and converts them to appropriate values before concatenating them into a comma delimited string. This string is then passed through the VISA resource to the

Arduino, and the operation mode is passed to future switch-case checks to implement various functions.



If the mode selected is ‘Zero System’, ‘Infuse’, ‘Withdraw’, or ‘Oscillate’, the next stage is bypassed. If the mode selected is ‘Prime’, the software will update the ‘Prime’ text display on the front panel. The script will display the next step to be performed and waits for the user to toggle the next step button before sending a digital ‘high’ to the Arduino to

execute the next step in the priming process. The Arduino manages the actual priming steps. If the mode selected is ‘PWM’, the script waits for the Arduino to complete a few variable calculations and a handshake verification confirming receipt of data. The code then writes the calculated values passed from the Arduino to the displays on the front panel.



Finally, if either the ‘Record Pressure’ or ‘Record Absorbance’ flags are true, the script checks the VISA for content every five milliseconds (frequency of 200 Hz, 4 times faster than the collection rate of the Arduino to avoid aliasing and data loss). When content is found, the loop checks to make sure the correct number of bytes are present to avoid data chopping. This string is then read and passed to the next stage. The string is checked to determine if the data collection has terminated. If it has not been terminated, the script parses the data contained in the string, appends the data to the respective vectors as a function of time, and plots them to the self-adjusting graphs on the front panel. When the string reads ‘endthisrun’ the run is terminated, variables are reset as necessary, and the system returns to waiting for user input.

A.5 Modelling of Periodic Backflush

A.5.1 Software

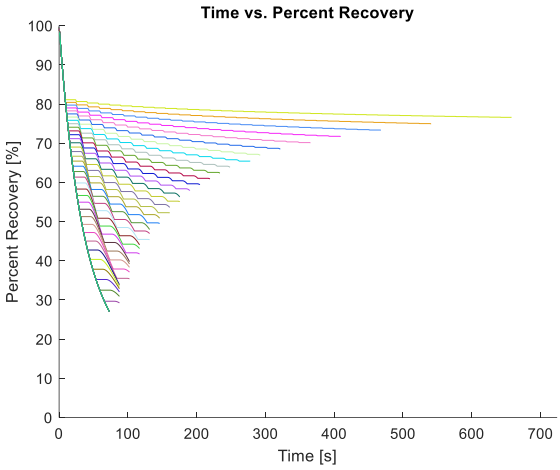
MATLAB

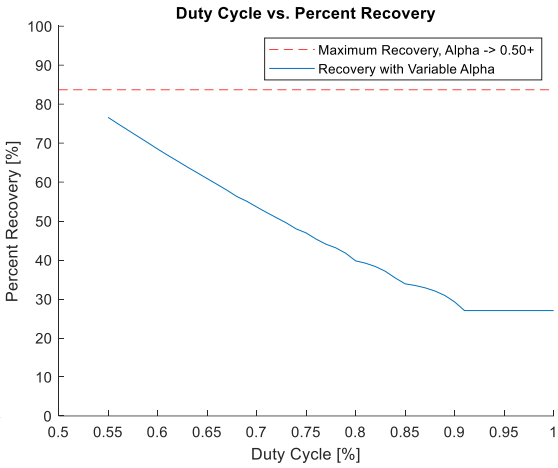
A.5.2 Output

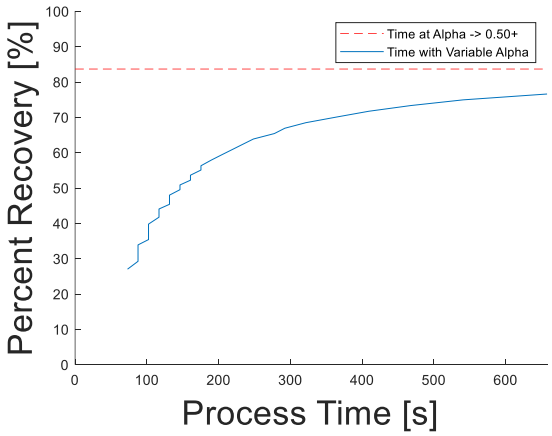
This script is split into two parts: a plotting function and a modelling function. Calling the plotting function establishes a vector of values for duty cycle and iterates through each duty cycle to determine the time course data for concentration of each. These values are stored into different vectors and plots for time versus recovery, duty cycle versus percent recovery, process time vs percent recovery, process time vs duty cycle, and duty cycle versus recovery percentage divided by process time are generated.

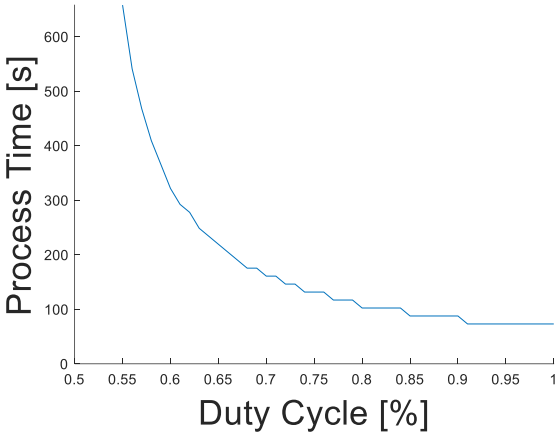
A.5.3 Description of Function

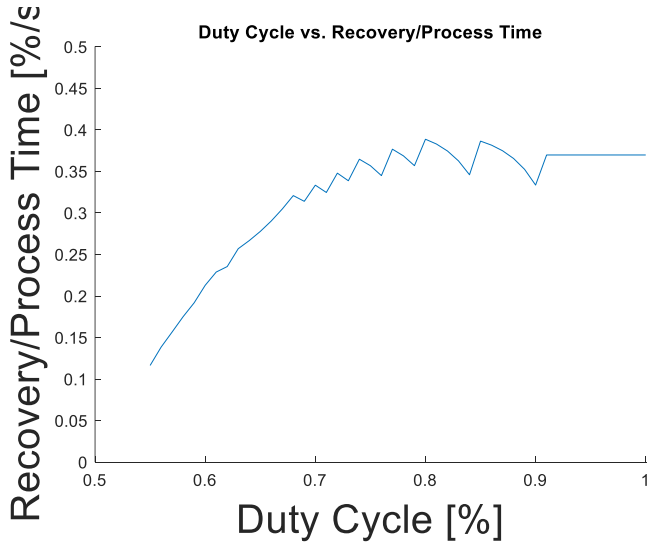
The script plotFilt iteratively calls expFilt for each duty cycle to be analysed. The script passes both the duty cycle value as well as backflush volume for the cycle being analysed. The function expFilt then defines certain constants, iteratively calculates parameters defined in Chapter 4.4.3, and returns the time, flux, filtrate mass, and concentration vectors for plotting.


Time vs. Percent Recovery
Percent Recovery [%]
Time [s]


Duty Cycle vs. Percent Recovery
Maximum Recovery, Alpha -> 0.50+
Recovery with Variable Alpha
Percent Recovery [%]
Duty Cycle [%]


Time at Alpha -> 0.50+
Time with Variable Alpha
Percent Recovery [%]
Process Time [s]


Process Time [s]
Duty Cycle [%]



A.5.4 Code

PlotFilt

```
function plotFilt
% Matthew Leipner, Aaron Enten
% Nov 7, 2017

%Set constant backflush volume
vR = 1E-4;                %[m^3]      Constant backflush volume

%Plot end concentration vs. Alpha
alpha = 0.55:0.01:1.0;

figure
hold on

%Calculate successive alphas
finalCVec = [];
finalTVec = [];
tVec = {};
CVec = {};
h = waitbar(0,'Initializing waitbar...');
for i = 1:length(alpha)
```

```
    waitbar(i./length(alpha),h,sprintf('Currently calculating alpha of %d', alpha(i)))
    [t,~,~,~,C] = expFilt(alpha(i),vR);
    tVec = [tVec, t];
    CVec = [CVec, C];
    finalCVec = [finalCVec, C(end)];
    finalTVec = [finalTVec, t(end)];
    plot(tVec{i},CVec{i},'Color',rand(1,3))
    labeln = num2str(alpha(i));
    %annotation('textarrow',[.1 .2],[.1 .2],'String','my text')
end
waitbar(i./length(alpha),h,'Calculations complete')

axis([0 finalTVec(1).*1.1 0 100])
title('Time vs. Percent Recovery')
xlabel('Time [s]')
ylabel('Percent Recovery [%]')

%Plot alpha vs. final concentration
figure
hold on
plot(linspace(0.5,length(alpha)),zeros(1,100)+83.6740,'r--')
plot(alpha,finalCVec)
axis([0.5 1 0 100])
close(h)
title('Duty Cycle vs. Percent Recovery')
xlabel('Duty Cycle [%]')
ylabel('Percent Recovery [%]')
legend('Maximum Recovery, Alpha -> 0.50+','Recovery with Variable Alpha')

%Plot final time vs. final concentration
figure
hold on
plot(linspace(0,finalTVec(1)),zeros(1,100)+83.6740,'r--')
plot(finalTVec,finalCVec)
axis([0 finalTVec(1) 0 100])
%title('Process Time vs. Percent Recovery')
xlabel('Process Time [s]','fontsize',24)
ylabel('Percent Recovery [%]','fontsize',24)
legend('Time at Alpha -> 0.50+','Time with Variable Alpha')

%Plot alpha vs. process time
figure
```

```
hold on
plot(alpha,finalTVec)
axis([0.5 1 0 finalTVec(1)])
%title('Duty Cycle vs. Process Time')
xlabel('Duty Cycle [%]','fontsize',24)
ylabel('Process Time [s]','fontsize',24)
%legend('Time at Alpha -> 0.50+','Time with Variable Alpha')

%Plot alpha vs. Recovery/Process Time
figure
hold on
plot(alpha,finalCVec./finalTVec)
axis([0.5 1 0 0.5])
title('Duty Cycle vs. Recovery/Process Time')
xlabel('Duty Cycle [%]','fontsize',24)
ylabel('Recovery/Process Time [%/s]','fontsize',24)

% for i = 1:length(alpha)
%     xlswrite('TheoreticalData.xlsx',CVec(i),'Sheet2',sprintf('A%d',i));
% end

end
```

ExpFilt

```
function [tVec,J,mP,V,C] = expFilt(alpha,vR)
% Matthew Leipner, Aaron Enten
% Aug 3, 2017

%Inlet Parameters
percA = 50;    %[%]Percent of beads which will pass through membrane (bead "A")
%vT = 1E-6;             %[m^3]     Total volume to be processed
vT = 1E-3;              %[m^3]     Total volume to be processed
nCalc = 1E2;   %[-] Number of calculations to complete per cycle (more provides
smoother curve)

%Define Constants
%Fluid Flow
%Q0 = 7.57313E-08;          %[m^3/s]    Initial volumetric flow rate (Flux*Area)
Q0 = 1.3678E-5;         %[m^3/s]    Initial volumetric flow rate (Flux*Area)
%Particle Flow
```

```
Am = 2.8E-4;              %[m^2]     Effective membrane area
dP = 310E3;               %[Pa]      Driving force (transmembrane pressure drops)

                %***CITE SYRINGE FILTER MANUFACTURER***%[7]
Cb0 = 0.5;                %[kg/m^3]  Bulk concentration
J0 = Q0*Cb0;              %[kg/s]    Initial particle flow rate = (volume flow rate)*(particle
concentration)
mu = 9.0925e-04;          %[Pa*s]    Viscosity of water (analogous to PBS solution)
Rm = dP.*Am./(mu.*J0);%[m^-1]    Resistance of clean membrane, modified from
Darcy's law
                %***CITE TRANSPORT TEXT FOR DARCY'S LAW***%[120]

%Constants from Experimental Data
a  = 4.1;  %[m^2/kg]   Pore blockage parameter, membrane area blocked/unit mass bead
convected to membrane surface
fR = 2.4E12;%[m/kg]   Combined f'R', rate of increase of bead layer resistance with time
Rp0 = 4E11;%[m^-1]    Initial resistance of bead deposit
 %*CITE HO & ZYDNEY FOR EXPERIMENTAL VALUES USED IN MODEL*%[103]

%Calculate time intervals for exp-decay forward, low-resistance forward, and backflush
if alpha == 1
   tVec = 0:(vT./Q0)./nCalc:vT./Q0;    %[s]  Single calculation time vector for alpha of 1
else
   vF = vR.*alpha./(1-alpha);         %[m^3]  Volume pushed forward
   tR = vR./Q0;                       %[s]    Time of backflush and low-resistance forward flow
   tF = vF./Q0;                       %[s]    Time of exp-decay forward flow
   if vF >= vT
      critAlpha = 1/(1+tR*Q0*(tF-tR)/(vT*(tF-tR)));
      %sprintf('inlet alpha of %d is greater than or equal to critical alpha of %d, will
calculate as alpha = 1',alpha,critAlpha)
      alpha = 1;                          %Sets function to calculate based on alpha = 1
      tVec = 0:(vT./Q0)./nCalc:vT./Q0; %[s] Single calculation time vector for alpha of 1
   else
      rawN = (vT-vF)./(vF-vR);    %[Cycles] Raw number of cycles required to reach
desired volume processed
      N = ceil(rawN);           %[Cycles] Rounded number of cylces to be completed in
calculations (rounded up)
      fCycle = mod(rawN,1);       %[Cycles] Fraction of forward flow to be completed
after final calculations
   end
end
```

```
%Initialize Flux calculations
J = [J0];                %[kg/s]     mass flow rate of particles through membrane vector
mP = [0];                %[kg]       mass of particles fluxed through membrane vector
V = [0];                 %[m3]       permeate volume vector
C = [Cb0];               %[kg/m3]    particle concentration vector
if alpha == 1            %Runs single forward flow if alpha = 1
   Cb = Cb0;
   RpVec = (Rm+Rp0).*sqrt(1+(2.*fR.*dP.*Cb.*tVec./(mu.*(Rm+Rp0).^2)))-Rm;
   J  = J0.*exp(-a.*dP.*Cb.*tVec./(mu.*Rm)+(Rm./(Rm+RpVec)).*(1-exp(-
a.*dP.*Cb.*tVec./(mu.*Rm))));
   V  = Q0.*tVec;

   %Calculate total permeate particle mass (used for concentration)
   mP = zeros(1,length(tVec));
   for i = 2:length(tVec)
      mP(i) = trapz(tVec(1:i),J(1:i));
   end
   C  = mP./V;
else
%Iterate through number of cycles necessary to process desired volume (appends
calculated vectors to total vectors at each step)
    for i = 1:N
       %*****************************************************%
       %Forward-Flush Calculations
       %*****************************************************%
       if i == 1
          Cb = Cb0;
          tVec = 0:tF./nCalc:tF;
          RpVec = (Rm+Rp0).*sqrt(1+(2.*fR.*dP.*Cb.*tVec./(mu.*(Rm+Rp0).^2)))-Rm;
          J  = J0.*exp(-a.*dP.*Cb.*tVec./(mu.*Rm)+(Rm./(Rm+RpVec)).*(1-exp(-
a.*dP.*Cb.*tVec./(mu.*Rm))));
%{
          P  = P0.*exp(-a.*DF.*Cb.*tVec./(mu.*Rm)-(a.*exp(-
a.*(mu.*(Rm+Rp0)^2+2.*Cb.*DF.*fR.*tVec)./(2.*fR.*mu.*Rm)).*(Rm+Rp0)...
                .*sqrt((pi./2)+(Cb.*DF.*fR.*pi.*(tVec))./(mu*(Rm+Rp0)^2))...
                .*(-1+erf(sqrt(-
a.*(mu.*(Rm+Rp0)^2+2.*Cb.*DF.*fR.*(tVec)).*log(2.71828)./(fR.*mu.*Rm))./sqrt(2))..
.
                ./(fR.*sqrt(-
a.*(mu.*(Rm+Rp0)^2+2.*Cb.*DF.*fR.*(tVec)).*log(2.71828)./(fR.*mu.*Rm)))))...
                -(a.*exp(-
a.*(mu.*(Rm+Rp0)^2+2.*Cb.*DF.*fR.*tVec)./(2.*fR.*mu.*Rm)).*(Rm+Rp0)...
```

```
            .*sqrt((pi./2))...
            .*(-1+erf(sqrt(-
a.*(mu.*(Rm+Rp0)^2).*log(2.71828)./(fR.*mu.*Rm))./sqrt(2))...
            ./(fR.*sqrt(-a.*(mu.*(Rm+Rp0)^2).*log(2.71828)./(fR.*mu.*Rm))))));
%}
        V  = Q0.*tVec;

        %Calculate total permeate particle mass (used for concentration)
        mP = zeros(1,length(tVec));
        for j = 2:length(tVec)
           mP(j) = trapz(tVec(1:j),J(1:j));
        end
        C  = mP./V;

        %Calculate fractional fouling cleared in backflush
        [~,t2] = size(tVec);              %Time at which exp-decay forward flow completed
        Rp = (1-alpha).*RpVec(end); %Changes starting fouling parameter to fraction
    else
        [~,t1] = size(tVec);              %Time at which low-res forward flow completed
        tCalc  = 0:(tF-tR)./nCalc:(tF-tR);
        RpCalc = (Rm+Rp).*sqrt(1+(2.*fR.*dP.*Cb.*tCalc./(mu.*(Rm+Rp).^2)))-Rm;
        PCalc  = J(end).*exp(-a.*dP.*Cb.*tCalc./(mu.*Rm)+(Rm./(Rm+RpCalc)).*(1-
exp(-a.*dP.*Cb.*tCalc./(mu.*Rm))));
        vCalc  = Q0.*tCalc;
        tVec   = [tVec, tVec(end)+tCalc];
        RpVec     = [RpVec, RpCalc];
        J      = [J, PCalc];
        V      = [V, V(end)+vCalc];

        %Calculate total permeate particle mass (used for concentration)
        mPCalc = zeros(1,length(tCalc));
        for j = 1:length(tCalc)
           mPCalc(j) = trapz(tVec(1:t1+j),J(1:t1+j));
        end
        mP = [mP, mPCalc];
        C  = [C, mPCalc./V(t1+1:end)];

        %Calculate fractional fouling cleared in backflush
        [~,t2] = size(tVec);       %Time at which exp-decay forward flow completed
        Rp = (1-alpha).*RpVec(end);   %Changes starting fouling parameter to fraction
    end
```

```
        %**************************************%
        %Back-Flush Calculations
        %**************************************%
        tCalc = 0:tR./nCalc:tR;
        J  = [J, zeros(1,length(tCalc))-C(end).*Q0];
        V  = [V, V(end)-Q0.*tCalc];
        mP = [mP,zeros(1,length(tCalc))+mP(end)+J(end).*tCalc];
        C  = [C, zeros(1,length(tCalc))+C(end)];
           %Permeate concentration remains const during backflush
        tVec = [tVec, tVec(end)+tCalc];
        RpVec = [RpVec, zeros(1,nCalc+1)+RpVec(end)];

        %*******************************%
        %Low resistance Forward-Flux Calculations (same volume as backflush period,
ignores cake resistance)
        %*******************************%
        tCalc = 0:tR./nCalc:tR;
        J  = [J, zeros(1,length(tCalc))+C(end).*Q0];
%Subtract P(end) as the value will be negative, resulting in a positive P
        V  = [V, V(end)+Q0.*tCalc];
        mP = [mP,zeros(1,length(tCalc))+mP(end)+J(end).*tCalc];
        C  = [C, zeros(1,length(tCalc))+C(end)];
%Permeate concentration remains const during backflush
        tVec = [tVec, tVec(end)+tCalc];
        RpVec = [RpVec, zeros(1,nCalc+1)+RpVec(end)];
        PStart = (1-alpha).*(J0-J(end))+J(end);  %Update starting flux based on irr. fouling
    end
    %*********************************%
    %Compute final fractional cycle
    %*********************************%
    if fCycle>0
    [~,t3] = size(tVec);          %Time at which low-res forward flow completed
    tCalc  = 0:fCycle.*(tF-tR)./nCalc:fCycle.*(tF-tR);
    RpCalc = (Rm+Rp).*sqrt(1+(2.*fR.*dP.*Cb.*tCalc./(mu.*(Rm+Rp).^2)))-Rm;
    PCalc  = J(end).*exp(-a.*dP.*Cb.*tCalc./(mu.*Rm)+(Rm./(Rm+RpCalc)).*(1-exp(-
a.*dP.*Cb.*tCalc./(mu.*Rm))));
    vCalc  = Q0.*tCalc;
    tVec   = [tVec, tVec(end)+tCalc];
    RpVec     = [RpVec, RpCalc];
    J      = [J, PCalc];
    V      = [V, V(end)+vCalc];
```

```
    %Calculate total permeate particle mass (used for concentration)
    mPCalc = zeros(1,length(tCalc));
    for j = 1:length(tCalc)
        mPCalc(j) = trapz(tVec(1:t3+j),J(1:t3+j));
    end
    mP = [mP, mPCalc];
    C  = [C, mPCalc./V(t3+1:end)];
    end
end
C = C./Cb0*100;
%plot(tVec,C)
%Plotting for Paper
%{
figure
hold on
plot(tVec,mP./max(mP),'r-')
plot(tVec,J./max(J),'b')
xlabel('Time [s]')
ylabel('Fraction of maximum [%]')
legend('Permeate solute mass [kg/kg]','Solute Flux [kg/m2s]')
plot(linspace(0,tVec(end)),zeros(1,100)),'k')
axis([0 tVec(end) -1 1])
%}
end
```

A.6 Model for handling waveform packets

A.6.1 Software

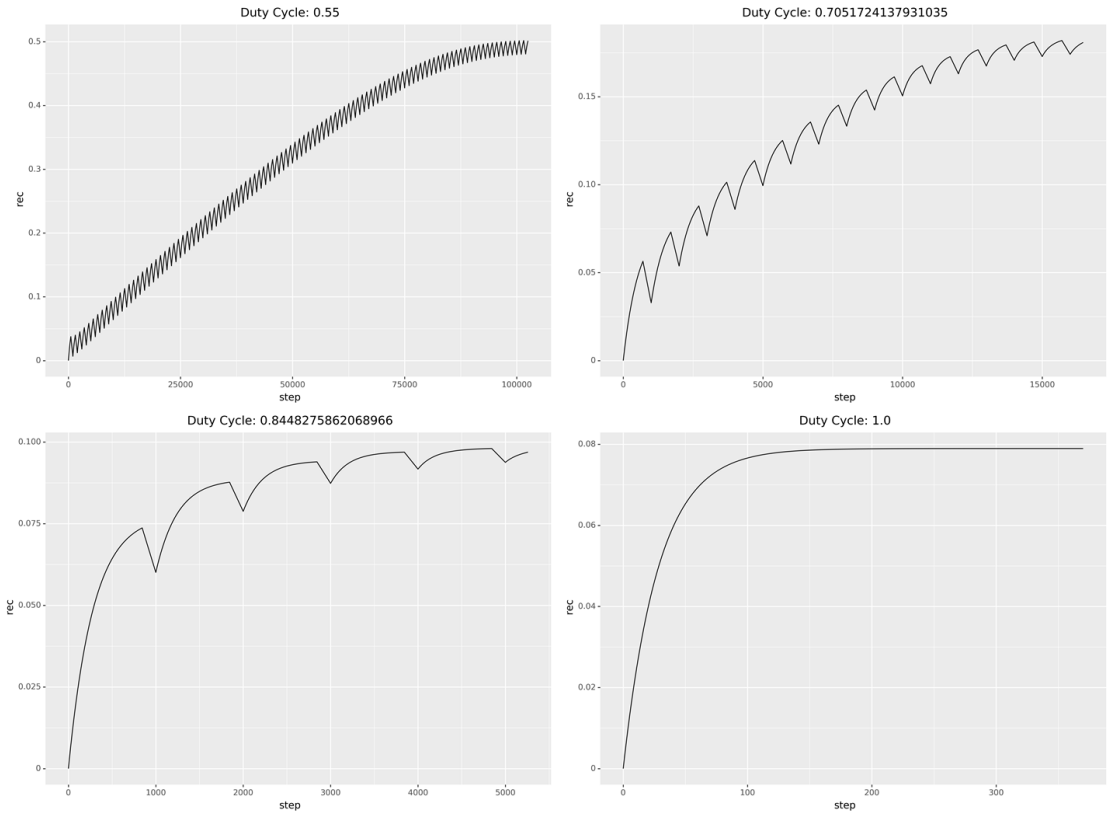
Python

A.6.2 Output

These scripts produce two main results. The initial goal is to determine blocking and clearance rate constants, and the second is to generate recovery percentage calculations for

the combination of a series of backflush volumes and duty cycles using the previously calculated rate constants.

The first script regresses the recovery percentage data for interpolated points against the results from FBF PWM recovery percentage. This results in Figure 28 in the main text. Recovery percentage is calculated as the end point of a recovery percentage vs time curve for a fixed volume of bulk solution processed. The script then provides a csv document containing the data for all curves calculated using the final regression results.



The second output of the script is a 3D surface plot of recovery percentage as a function of a range of duty cycles and backflush volumes, color mapped by processing time, Figure 30.

A.6.3 Description of Function

Th software works by creating a waveform packet for a single period of an expected wave. The script uses the flow rate and timing information of this wave to increment through time steps, calculating cake mass, solute flux, solvent flux, cake clearance, remaining bulk, bulk concentration, processing time, and recovery percentage as well as other critical variables. Once data for all time points of all calculation are determined, they are both stored and plotted as appropriate in csv and appropriate photo file formats.

A.6.4 Code

```
import squave.model as model
import squave.plots as plots
import squave.wave as wave
import squave.fitting as fitting
```

```
import numpy as np
def squareWave(qF:float,qR:float,dc:float,waveLength:float,numPoints:float)->(float,list):
    """Create a list (suitable for the Q argument of squave()) representing a square wave
    qF is the forward flow rate
    qR is the reverse flow rate
    dc is the duty cycle
    waveLength is the total length (in seconds) of one cycle
    numPoints is the number of time steps"""
    tf=waveLength*dc #time flowing forward
```

```
    tr=waveLength*(1-dc) #time flowing backwards
    tList,ts=np.linspace(0,waveLength,numPoints,retstep=True)
    q=[qF if t<=tf else qR for t in tList]
    return ts,q

def swFixedReverse(qF:float,qR:float,dc:float,Vr:float,numPoints:float)->(float,list):
    """Create a list (suitable for the Q argument of squave()) representing a square wave
    qF is the forward flow rate
    qR is the reverse flow rate
    dc is the duty cycle
    vR is the amount of volume to backflush per cycle
    numPoints is the number of time steps"""
    tR=abs(Vr/qR) #reverse time
    tTotal=(1/(1-dc))*tR #total wavelength
    tF=tTotal*dc #time forward
    tList,ts=np.linspace(0,tTotal,numPoints,retstep=True)
    q=[qF if t<=tF else qR for t in tList]
    return ts,q
```

```
import squave.model as sq
import squave.wave as wav
import numpy as np
import pandas as pd
import os
from plotnine import *
from mpl_toolkits.mplot3d import Axes3D
from mpl_toolkits.axes_grid1 import make_axes_locatable
import matplotlib as mpl
import matplotlib.pyplot as plt
from matplotlib import cm

expConditions={'Cb':2.5E9,
               'Cs':2.5E9,
                  'Vbig':2.05367571E-16,
                  'Vsmall':5.49959827E-18,
                  'Vb':1E-3}
```

```
swArgs=dict(qF=7.5E-5,qR=-7.5E-5,Vr=0.0000419152,numPoints=10000)
fitData={'rc': 24779563364.231182, 'ro': 16279.150700398039}

def plotRecoveryVsDC(saveDir:str):
    rec=[]
    dutyCycles=np.linspace(.55,1,30)
    for dc in dutyCycles:
        print(dc)
        if dc<1:
            ts,sw=wav.swFixedReverse(dc=dc,**swArgs)
        else:
            sw=[swArgs['qF']]
        #return ts,sw
        res=sq.squave(Q=sw,ts=ts,**fitData,**expConditions)
        plotTitle='Duty Cycle: '+str(dc)
        pd.DataFrame(res).to_csv(os.path.join(saveDir,plotTitle+'.csv'))
        print('plotting...')

(ggplot(pd.DataFrame(res),aes(x='step',y='rec'))+geom_path()+labs(title=plotTitle)).save(
filename=os.path.join(saveDir,plotTitle+'.png'),width=10,height=7,dpi=500)
        rec.append(res['rec'][-1])
    out=pd.DataFrame({'dc':dutyCycles,'rec':rec})
    out.to_csv(os.path.join(saveDir,'recVsDC.csv'))
    (ggplot(out,aes(x='dc',y='rec'))+geom_point()+labs(title='Recovery Percentage vs.
Duty
Cycle')).save(filename=os.path.join(saveDir,'recVsDC.png'),width=10,height=7,dpi=500)
    return out

def getSurfaceFrame(vR=np.linspace(.00004556,.00025,100),dc=np.linspace(.55,1,100)):
    swArgsLocal=swArgs.copy()
    del swArgsLocal['Vr']
    vROut=[]
    dcOut=[]
    fOut=[]
    recOut=[]
    for thisVr in vR:
```

```
            for thisDC in dc:
                print('thisVr: ',thisVr)
                print('thisDC: ',thisDC)
                if thisDC<1:
                    ts,q=wav.swFixedReverse(dc=thisDC,Vr=thisVr,**swArgsLocal)
                    thisF=1/((len(q)-1)*ts)
                else:
                    q=[swArgsLocal['qF']]
                    thisF=0
                res=sq.squave(Q=q,ts=ts,**expConditions,**fitData)
                thisRec=res['rec'][-1]
                vROut.append(thisVr)
                dcOut.append(thisDC)
                fOut.append(thisF)
                recOut.append(thisRec)
        return(pd.DataFrame(dict(vR=vROut,dc=dcOut,f=fOut,rec=recOut)))

def plotSurface(surfFrame:'DataFrame'):
        fig = plt.figure()
        ax = fig.add_subplot(111, projection='3d')
        surfMatrix=surfFrame.pivot('dc','vR','rec')
        fMatrix=surfFrame.pivot('dc','vR','f').to_numpy()
        vR,dc=np.meshgrid(surfMatrix.columns,surfMatrix.index)
        rec=surfMatrix.to_numpy()
        #fMatrix=np.log(fMatrix)
        my_col=cm.coolwarm(fMatrix)

        surf = ax.plot_surface(dc,vR, rec,
rcount=1000,ccount=1000,antialiased=False,facecolors=my_col)
        ax.set_xlabel('Duty Cycle')
        ax.set_ylabel('Reverse Volume (L)')
        ax.set_zlabel('Recovery')
        m=cm.ScalarMappable(cmap=cm.coolwarm)
        m.set_array(fMatrix)
        cb=fig.colorbar(m)
        cb.set_label('Frequency (Hz)')
```

```
    plt.show()
```

```
import math

def
squave(Q:list,ts:float,Cb:float,Cs:float,Vbig:float,Vsmall:float,Vb:float,rc:float,ro:float)-
>dict:
    """Run periodic backflush model: Q should be a list-like object giving the flow rate at
each time step, in L/s
    ts is the time step in seconds
    Cb is the initial concentration (in particles/L) of 'big' particles in the bulk
    Cs is the initial concentration (in particles/L) of 'small' particles in the bulk
    Vbig is the volume of the 'large' particles
    Vsmall is the volume of the 'small' particles
    Vb is the initial volume of the bulk (in L)
    rc is the rate constant dictating how fast permeate flux decays with aggregated material
    ro is the rate constant dictating how fast the filter clears with flushed volume"""
    Cp=0 #initial concentration of filtrate in the permeate
    Vp=0 #initial volume of the permeate
    smallCapturedForward=0 #number of 'small' particles initially trapped in the filter on
the bulk side
    smallCapturedReverse=0 #number of 'small' particles initially trapped in the filter on
the permeate side
    bigCapturedForward=0 #number of 'big' particles initially trapped in the filter
    perm=1 #initial permeability of the membrane
    numSmall=Cs*Vb #total number of 'small' particles
    maxSmallCapturedForward=0 #Max number of particles trapped on either side of the
filter at the end of a forward or reverse stroke
    maxSmallCapturedReverse=0
    maxBigCapturedForward=0
    vForward=0 #total amount of fluid processed on this forward stroke
    vReverse=0 #total amount of fluid processed on this reverse stroke
    rec=(Vp*Cp)/numSmall #recovery percentage
    out={'Cb':[Cb],
        'Cs':[Cs],
```

```
        'Vb':[Vb],
        'Cp':[Cp],
        'Vp':[Vp],
        'perm':[perm],
        'rec':[rec],
        'smallCapturedForward':[smallCapturedForward],
        'smallCapturedReverse':[smallCapturedReverse],
        'bigCapturedForward':[bigCapturedForward]} #initialize output lists
    while Vb>0: #run until we run out of bulk
        for q in Q: #for each timestep
            if Vb<=0:
                break
            vProcessed = abs(q*ts)  # amount of bulk volume processed during this time step
            if q>=0: #if we're moving fluid forward
                if vProcessed>Vb: #we are out of bulk fluid, just process the remainder
                    vProcessed=Vb
                vReverse=0 #reset reverse per stroke counter
                bigProcessed=Cb*vProcessed
                smallProcessed=Cs*vProcessed
                bigCapturedForward+=bigProcessed #all 'big' particles are captured
                smallCapturedForward+=(1-perm)*smallProcessed #small particles are
captured depending on permeability of filter
                maxSmallCapturedForward=smallCapturedForward
                maxBigCapturedForward=bigCapturedForward
                smallPassed=perm*smallProcessed #all small particles that are not captured go
through the filter
                numSmallPermeate=Cp*Vp #number of small particles in the permeate before
this time step
                Vp+=vProcessed #Calculate new Vp
                vForward+=vProcessed
                #recalculate how many small particles are trapped on the filtrate side
(exponentially approaches zero)
                newSmallCapturedReverse=maxSmallCapturedReverse*math.exp(-
1*ro*vForward)
                smallReleasedReverse=smallCapturedReverse-newSmallCapturedReverse
                smallCapturedReverse=newSmallCapturedReverse
```

```
            numSmallPermeate+=smallPassed+smallReleasedReverse
            Cp=numSmallPermeate/Vp #calculate new Cp
            Vb-=vProcessed #calculate new Vb

volCaptured=Vbig*bigCapturedForward+Vsmall*(smallCapturedForward+smallCapture
dReverse)
            perm=math.exp((-1)*rc*volCaptured)
            rec=(Vp*Cp)/numSmall #recovery percentage
            #record output values at this time step
            out['Cb'].append(Cb)
            out['Cs'].append(Cs)
            out['Vb'].append(Vb)
            out['Cp'].append(Cp)
            out['Vp'].append(Vp)
            out['perm'].append(perm)
            out['rec'].append(rec)
            out['smallCapturedForward'].append(smallCapturedForward)
            out['smallCapturedReverse'].append(smallCapturedReverse)
            out['bigCapturedForward'].append(bigCapturedForward)
        else: #we're flowing backwards
            #release all captured particles into the bulk
            vForward=0 #reset forward per stroke flow counter
            smallCapturedReverse+=Cp*vProcessed*(1-perm)
            maxSmallCapturedReverse=smallCapturedReverse
            smallPassed=Cp*vProcessed*perm
            numBigBulk=Cb*Vb #number of big particles in the bulk before this time step
            numSmallBulk=Cs*Vb #number of smallParticles in the bulk before this time
step
            Vp-=vProcessed #update bulk and permeate volumes
            Vb+=vProcessed
            vReverse+=vProcessed
            if Vp<0: raise Exception('Calculated permeate volume is less than 0, check Q
input')
            newSmallCapturedForward=maxSmallCapturedForward*math.exp(-
1*ro*vReverse)
```

```
            newBigCapturedForward=maxBigCapturedForward*math.exp(-
1*ro*vReverse)
            bigReleasedForward=bigCapturedForward-newBigCapturedForward
            bigCapturedForward=newBigCapturedForward
            smallReleasedForward=smallCapturedForward-newSmallCapturedForward
            smallCapturedForward=newSmallCapturedForward
            numBigBulk+=bigReleasedForward #recalculate Cb
            Cb=numBigBulk/Vb
            numSmallBulk+=smallReleasedForward+smallPassed #recalculate Cs
            Cs=numSmallBulk/Vb

volCaptured=Vbig*bigCapturedForward+Vsmall*(smallCapturedForward+smallCapture
dReverse)
            perm=math.exp((-1)*rc*volCaptured)
            rec=(Vp*Cp)/numSmall #recovery percentage
            # record output values at this time step
            out['Cb'].append(Cb)
            out['Cs'].append(Cs)
            out['Vb'].append(Vb)
            out['Cp'].append(Cp) #Cp will not change when running backwards
            out['Vp'].append(Vp)
            out['perm'].append(perm)
            out['rec'].append(rec)
            out['smallCapturedForward'].append(smallCapturedForward)
            out['smallCapturedReverse'].append(smallCapturedReverse)
            out['bigCapturedForward'].append(bigCapturedForward)

    out['step']=list(range(len(out['rec'])))
    return out
```

```
import squave.model as sq
import squave.wave as wav
import pandas as pd
import scipy
import numpy as np
```

```
from statistics import median

swArgs=dict(qF=7.5E-5,qR=-7.5E-5,Vr=0.0000419152,numPoints=1000)
expConditions={'Cb':2.5E9,
            'Cs':2.5E9,
               'Vbig':2.05367571E-16,
               'Vsmall':5.49959827E-18,
               'Vb':1E-3}
def fitToData(dataPath:str):
   data=pd.read_csv(dataPath)
   data=pd.DataFrame(data.stack()).reset_index()
   data=data.iloc[:,1:]
   data.columns=['dc','rec']
   data=data.loc[data.loc[:,'dc']!='Duty Cycle',:]
   data.loc[:,'dc']=[int(i)/100 for i in data.loc[:,'dc']]
   data.loc[:,'rec']=[r/100 for r in data.loc[:,'rec']]
   data=data.groupby('dc').agg(median).reset_index()
   def func(dc,rc,ro):
'''dc is a list of duty cycles, return the corresponding list of recovery percentages'''
      out=[]
      print('rc:{}, ro:{}'.format(rc,ro))
      for d in dc:
         if d<1:
            ts,q=wav.swFixedReverse(dc=d,**swArgs)
         else:
            q=[swArgs['qF']]
         sqData=sq.squave(Q=q,ts=ts,rc=rc,ro=ro,**expConditions)
         out.append(sqData['rec'][-1])
      print(out)
      return out

fit=scipy.optimize.curve_fit(func,list(data.loc[:,'dc']),list(data.loc[:,'rec']),p0=(106897288
7,43791),bounds=(0,np.inf),maxfev=1000000)
   return {'rc':fit[0][0],'ro':fit[0][1]}
```

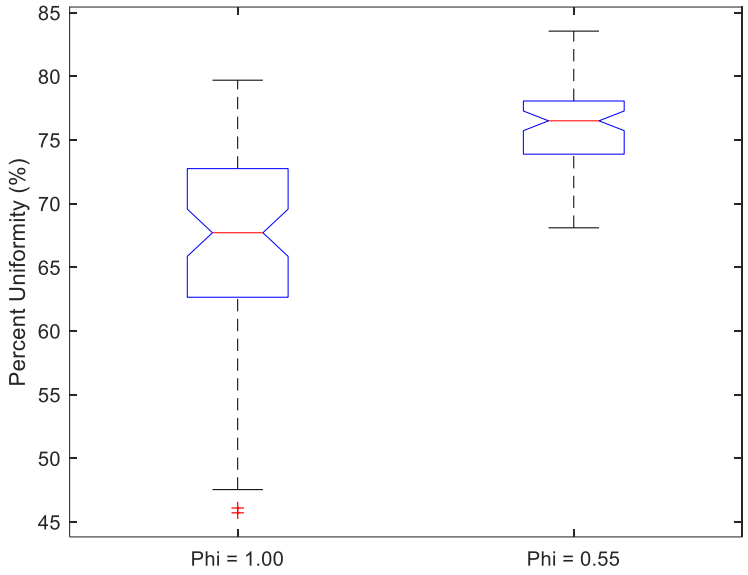
A.7 Percent Uniformity from Intensity Images

A.7.1 Software

MATLAB

A.7.2 Output

The purpose of this script is to take in two images for different duty cycle seeding experimental sets and analyse them line-by-line. The script will generate plots of the line-by-line intensity as well as surface plots mapped by intensity. The script will also output a box plot with notch on for percent uniformity distributions.



A.7.3 Description of Function

The script requires that two images named ‘Infusion’ and ‘PWM’ be in the working directory. It imports these images and stores the greyscale images into matrices for each image. The script calculates average row intensity and standard deviation and uses these to determine the row-by-row percent uniformity. It then passes this information on to calculate and plot the intensities as line plots and surfaces for each duty cycle.

A.7.4 Code

```
%% Reset workspace, clean command window, and Close all figures
close all
clear
clc

%% Redefine boxplot function to plot arrays of different lengths
col=@(x)reshape(x,numel(x),1);
boxplot2=@(C,varargin)boxplot(cell2mat(cellfun(col,col(C),'uni',0)),...
   cell2mat(arrayfun(@(I)I*ones(numel(C{I}),1),col(1:numel(C)),...
   'uni',0)),varargin{:});

%% Import Images and Store Grayscale Image sets
phiOne = imread('C:\Users\aente\Downloads\Uniformity\Infusion.png');
phiOneMsk = rgb2gray(phiOne);
phi55 = imread('C:\Users\aente\Downloads\Uniformity\PWM.png');
phi55Msk = rgb2gray(phi55);

%%Calculate the mean and stdev for each row of the images
iAveP1 = mean(phiOneMsk,2);
iAveP55 = mean(phi55Msk,2);
iStdP1 = std(double(phiOneMsk),0,2);
iStdP55 = std(double(phi55Msk),0,2);
```

%{ if calculating on cropped photos

```
%% Import Images and Store Grayscale Image sets
phiOne = imread('C:\Users\aente\Downloads\Uniformity\InfusionCropped.tif');
phiOneMsk = rgb2gray(phiOne);
```

```
phiOneMsk = double(phiOneMsk);
phiOneMsk(phiOneMsk==0) = NaN;
phi55 = imread('C:\Users\aente\Downloads\Uniformity\PWMcropped.tif');
phi55Msk = rgb2gray(phi55);
phi55Msk = double(phi55Msk);
phi55Msk(phi55Msk==0) = NaN;
%%Calculate the mean and stdev for each row of the images

iAveP1 = nanmean(phiOneMsk,2);
iAveP55 = nanmean(phi55Msk,2);
iStdP1 = std(phiOneMsk,0,2,'omitnan');
iStdP55 = std(phi55Msk,0,2,'omitnan');
```

```
%}

%% Calculate percent uniformity defined by Alverez-Barreto
UnifP1 = 100.*(1-iStdP1./iAveP1);
UnifP55 = 100.*(1-iStdP55./iAveP55);

%% Plot percent uniformity
U=cell(1,2);
U{1}=UnifP1;
U{2}=UnifP55;
boxplot2(U,'Notch','on','Labels',{'Phi = 1.00','Phi = 0.55'})
ylabel 'Percent Uniformity (%)'

%% Plot line-by-line intensity and calculate and plot surface progections
figure
hold on
plot(phiOneMsk')
title('\phi = 1.00')
ylabel 'Pixel Intensity'

figure
x = 0:size(phiOneMsk,2)-1;
y = 0:size(phiOneMsk,1)-1;
[X,Y] = meshgrid(x,y);
surf(X, Y, phiOneMsk)
colormap(jet)
xlabel 'Rows'
ylabel 'Columns'
zlabel 'Pixel Intensity'
```

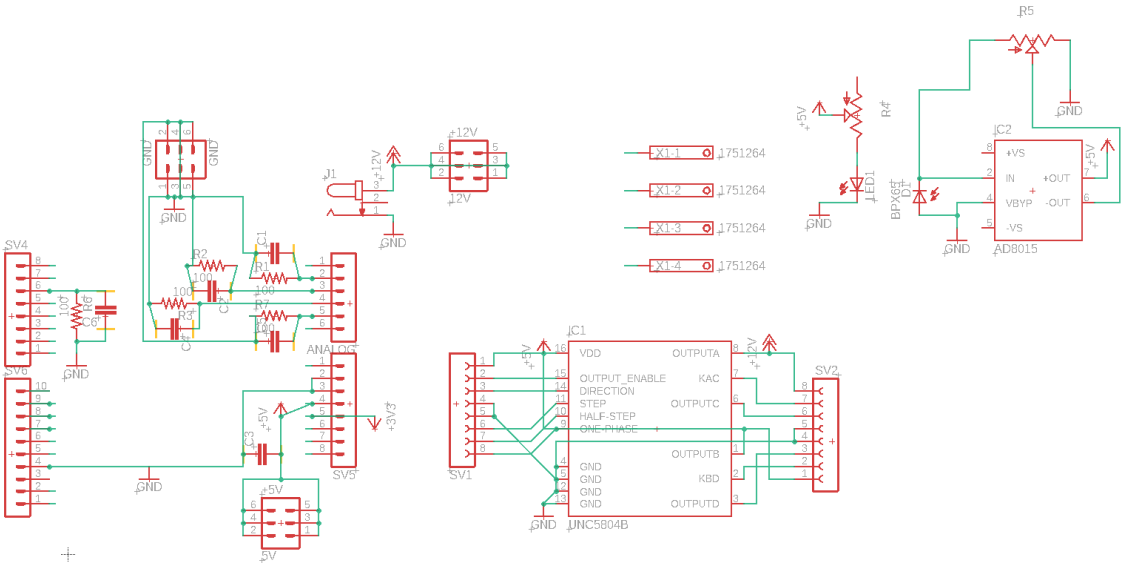
```
figure
hold on
plot(phi55Msk')
title('\phi = 0.55')
ylabel 'Pixel Intensity'

figure
x = 0:size(phi55Msk,2)-1;
y = 0:size(phi55Msk,1)-1;
[X,Y] = meshgrid(x,y);
surf(X, Y, phi55Msk)
colormap(jet)
xlabel 'Rows'
ylabel 'Columns'
zlabel 'Pixel Intensity'
```

APPENDIX B. HARDWARE

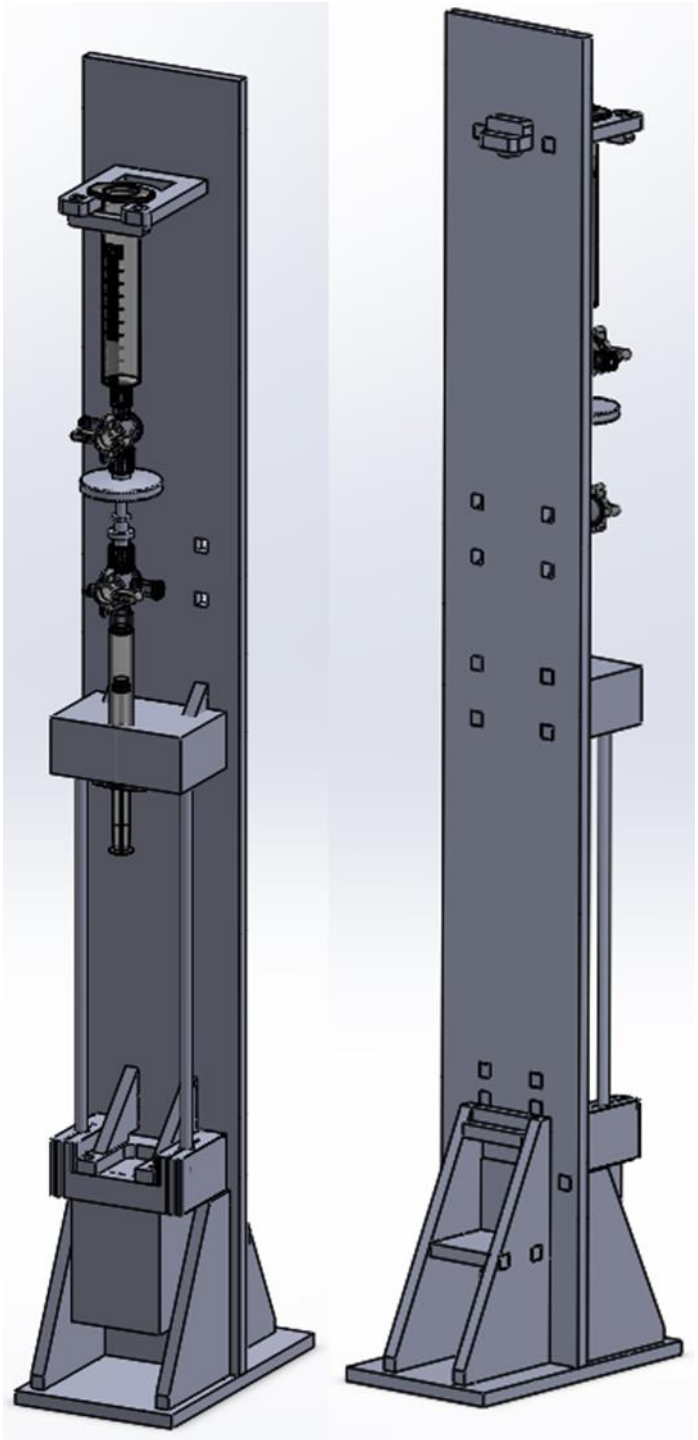
Appendix B is a repository of additional files relevant to hardware development generated over the course of this thesis. Each section of Appendix B will contain a title for the image presented in the first-level subheading, the hardware described by the title, and a description of the component or function as necessary. All files presented herein and those in the main thesis are available upon request of the author.

B.1 Motor Shield Circuit Diagram



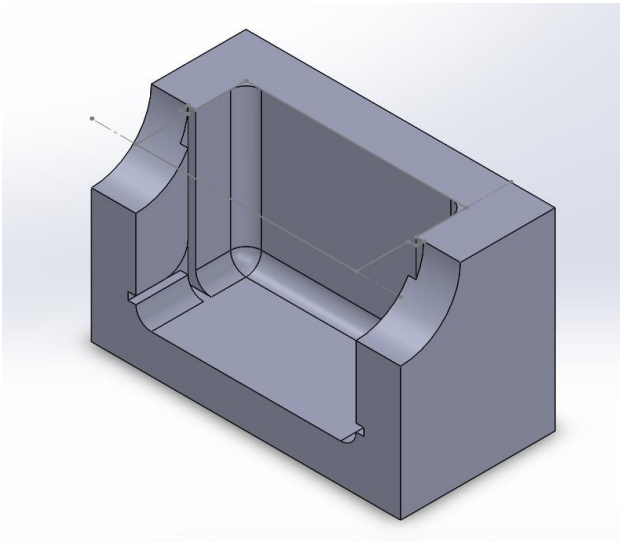
B.2 Hardware for Vertically Mounted Programmable Syringe Pump

Front and rear view of assembly structure used for vertical inverted PM programmable syringe pump.



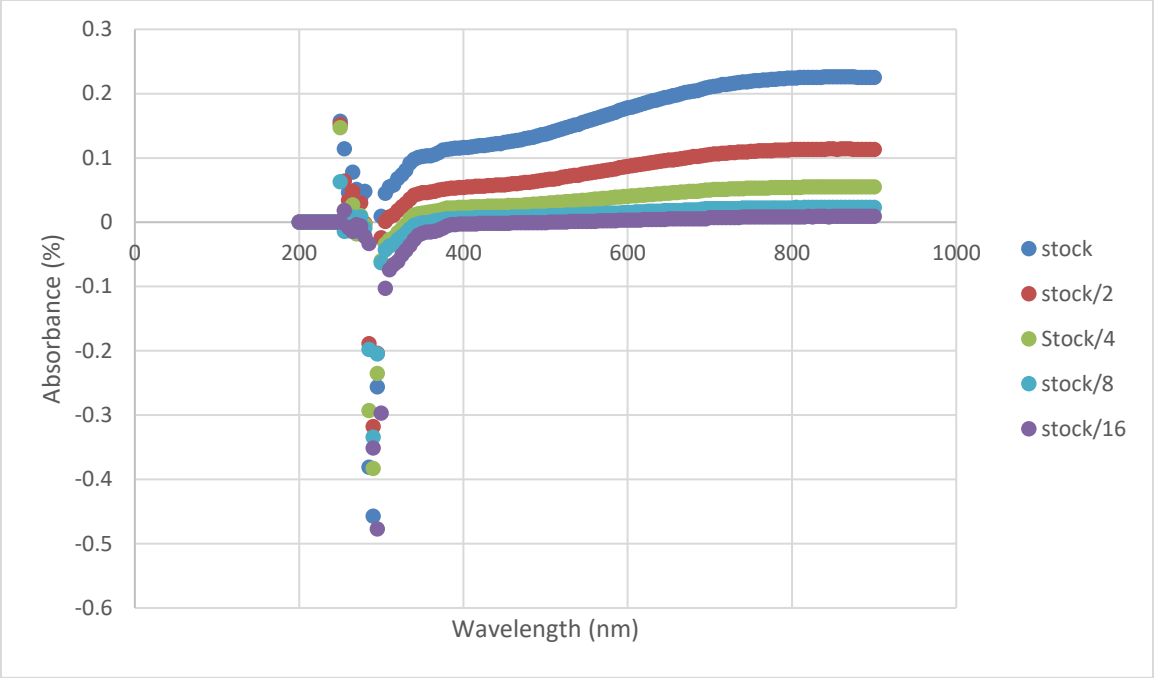
B.3 Thread catch detector

The thread catch detector was designed to move the actuator to an electrical contact and then step by step move away from the electrical contact until contact was broken. This was executed in both directions about one hundred times to calibrate and account for hysteresis in motor direction changes.



B.4 Spectrophotometry wavelength selection

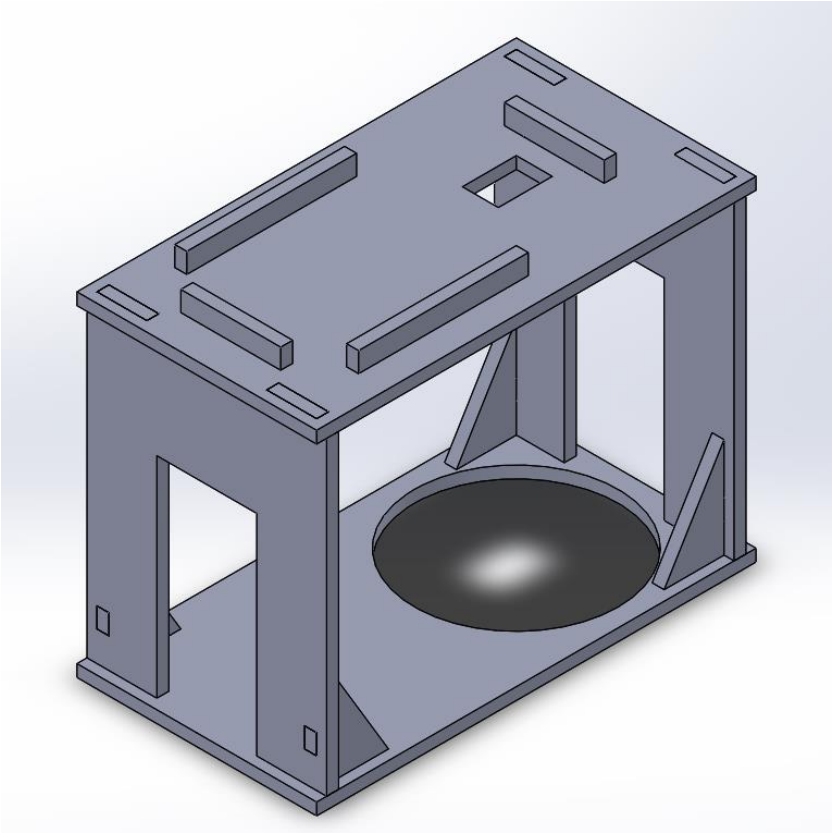
A future aspect of these experiments is to incorporate real time concentration calculation. The data below was used to determine the ideal wavelength absorbed most linearly responsively to polystyrene microspheres. The average double plot shows the average of each individual line divided by the line corresponding to half the previous line's concentration. Ideal absorption frequency would have an average absorption decrease of 2x for each dilution of previous solution by ½.



Frequency of 850 selected for experiments. ThorLabs LED851 and photodiode with corresponding center frequency from DigiKey selected. Excitation current produces 8 µA. Transimpedance Amplifier unit on motor shield is used to amplify and convert this signal.

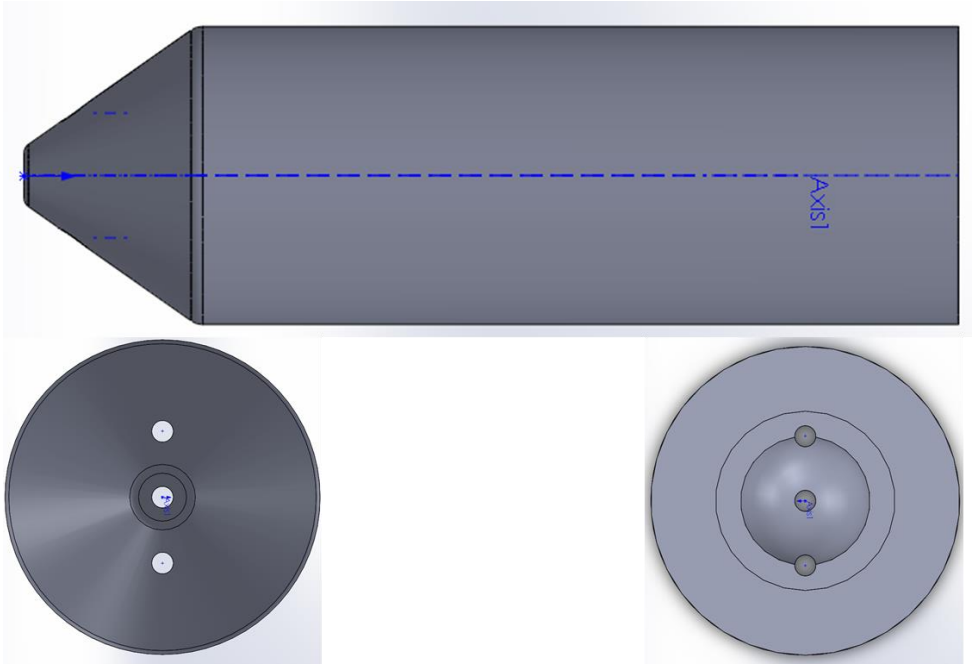
B.5 Cell phone Mounting Structure for Petri Dish Photography

Cellphone rests on top of system with alignment bars to ensure photograph orientation consistency. Concentric cutouts from lower portion allows for multiple sizes of petri dish. Future modification should include light shields to prevent or reduce reflections.



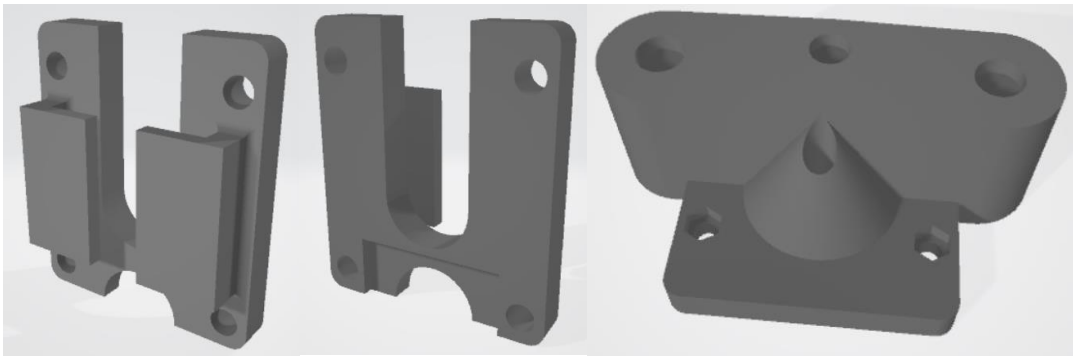
B.6 Culture tube to 50 mL Centrifuge Adapter

All samples were collected in FACS Culture Tubes. Quality Control requires centrifugation and resuspension of permeate. This adapter was created to enable the centrifugation of culture tubes to minimize losses associated with tube transfers.



B.7 Precision Syringe Modifications

Integration of a Hamilton gastight syringe was conducted to determine if precision syringe movement affected standard deviation of recovery percentage. Holster and actuator components were modified to account for larger syringe flanges and screw hole.



REFERENCES

1. Li H, Bertram CD, Wiley DE. Mechanisms by which pulsatile flow affects cross-flow microfiltration. *AIChE J*. 1998;44(9):1950-1961. doi:10.1002/aic.690440903

2. Blanpain-Avet P, Doubrovine N, Lafforgue C, Lalande M. The effect of oscillatory flow on crossflow microfiltration of beer in a tubular mineral membrane system - Membrane fouling resistance decrease and energetic considerations. *J Memb Sci*. 1999;152(2):151-174. doi:10.1016/S0376-7388(98)00214-2

3. Verwijst T, Baggerman J, Liebermann F, van Rijn CJM. High-frequency flow reversal for continuous microfiltration of milk with microsieves. *J Memb Sci*. 2015;494:121-129. doi:10.1016/j.memsci.2015.07.048

4. Czekaj P, Opez FL, G Uell C. Membrane fouling by turbidity constituents of beer and wine: characterization and prevention by means of infrasonic pulsing. http://ac.els-cdn.com/S0260877400001813/1-s2.0-S0260877400001813-main.pdf?_tid=eeb80a58-76e1-11e7-bf6c-00000aacb35d&acdnat=1501610021_69643e854675ab17974654817d9111ea. Accessed August 1, 2017.

5. Shields CW, Ohiri KA, Szott LM, López GP. Translating microfluidics: Cell separation technologies and their barriers to commercialization. *Cytom Part B Clin Cytom*. 2017;92(2):115-125. doi:10.1002/cyto.b.21388

6. Shields Iv CW, Ohiri KA, Szott LM, Opez GPL. Translating Microfluidics: Cell Separation Technologies and Their Barriers to Commercialization. doi:10.1002/cyto.b.21388

7. *Filtration, Separation, and Detection Products: Technology and Support from Field to Lab*.; 2011. https://us.vwr.com/assetsvc/asset/en_US/id/7037134/contents. Accessed March 2, 2019.

8. Bhave RR. *Cross-Flow Filtration*. http://ftp.feq.ufu.br/Luis_Claudio/Books/E-Books/Engineering/Fermentation And Biochemical Engineering Handbook/14077_07a.pdf. Accessed March 2, 2019.

9. Lee W, Tseng P, Di D. *Microtechnology for Cell Manipulation and Sorting*.; 2017. doi:10.1007/978-3-319-44139-9

10. Jones EA, Kinsey SE, English A, et al. Isolation and characterization of bone

marrow multipotential mesenchymal progenitor cells. *Arthritis Rheum*. 2002;46(12):3349-3360. doi:10.1002/art.10696

11. Scarberry KE, Dickerson EB, Zhang ZJ, Benigno BB, McDonald JF. Selective removal of ovarian cancer cells from human ascites fluid using magnetic nanoparticles. *Nanomedicine*. 2010;6(3):399-408. doi:10.1016/j.nano.2009.11.003

12. Phuoc NN, Ong CK. Electric field modulation of ultra-high resonance frequency in obliquely deposited [Pb(Mg1/3Nb2/3)O3]0.68-[PbTiO3]0.32(011)/FeCoZr heterostructure for reconfigurable magnetoelectric microwave devices. *Appl Phys Lett*. 2014;105(2):022905. doi:10.1063/1.4890411

13. Gorkin R, Park J, Siegrist J, et al. Centrifugal microfluidics for biomedical applications. *Lab Chip*. 2010;10(14):1758-1773. doi:10.1039/b924109d

14. Sethu P, Sin A, Toner M. Microfluidic diffusive filter for apheresis (leukapheresis). *Lab Chip*. 2006;6(1):83-89. doi:10.1039/b512049g

15. Pamme N. Continuous flow separations in microfluidic devices. *Lab Chip*. 2007;7(12):1644. doi:10.1039/b712784g

16. Wang G, Crawford K, Turbyfield C, Lam W, Alexeev A, Sulchek T. From chip-in-a-lab to lab-on-a-chip: towards a single handheld electronic system for multiple application-specific lab-on-a-chip (ASLOC). 2014;15. doi:10.1039/c4lc01150c

17. Wang G, Mao W, Byler R, et al. Stiffness dependent separation of cells in a microfluidic device. *PLoS One*. 2013;8(10):e75901. doi:10.1371/journal.pone.0075901

18. Sajeesh P, Ashis @bullet, Sen K. Particle separation and sorting in microfluidic devices: a review. 2013. doi:10.1007/s10404-013-1291-9

19. Wu Z, Willing B, Bjerketorp J, Jansson JK, Hjort K. Soft inertial microfluidics for high throughput separation of bacteria from human blood cells. *Lab Chip*. 2009;9(9):1193-1199. doi:10.1039/b817611f

20. Andersson H, van den Berg A. Microfluidic devices for cellomics: a review. *Sensors Actuators B Chem*. 2003;92(3):315-325. doi:10.1016/S0925-4005(03)00266-1

21. Wang X, Papautsky I. Size-based microfluidic multimodal microparticle sorter. *Lab Chip*. 2015. doi:10.1039/C4LC00803K

22. CHEN X, CUI D, LIU C, LI H. Microfluidic chip for blood cell separation and

collection based on crossflow filtration. *Sensors Actuators B Chem*. 2008;130(1):216-221. doi:10.1016/j.snb.2007.07.126

23. Gironès i Nogué M, Akbarsyah IJ, Bolhuis-Versteeg LAM, Lammertink RGH, Wessling M. Vibrating polymeric microsieves: Antifouling strategies for microfiltration. *J Memb Sci*. 2006;285(1):323-333. doi:10.1016/j.memsci.2006.09.001

24. Ji HM, Samper V, Chen Y, Heng CK, Lim TM, Yobas L. Silicon-based microfilters for whole blood cell separation. *Biomed Microdevices*. 2008;10(2):251-257. doi:10.1007/s10544-007-9131-x

25. Yoon Y, Kim S, Lee J, et al. Clogging-free microfluidics for continuous size-based separation of microparticles. *Sci Rep*. 2016;6(May):26531. doi:10.1038/srep26531

26. Legallais C, Moriniere P, Wojcicki JM, et al. A High Selectivity Cascade Filtration Technique for LDL-Cholesterol and Lp(a) Removal. *Artif Organs*. 1995;19(9):887-895.

27. Diagnostic Consumables: Molecular Diagnostic Filters & Vents. http://www.porex.com/products/clinical-sciences/ivd-molecular-diagnostics/. Published 2017. Accessed September 15, 2017.

28. Enten A, Yang Y, Ye Z, et al. A Liquid-Handling Robot for Automated Attachment of Biomolecules to Microbeads. *J Lab Autom*. August 2015. doi:10.1177/2211068215601846

29. Czermak P, Nehring D, Wickramasinghe R. Membrane Filtration in Animal Cell Culture. In: Humana Press; 2007:397-420. doi:10.1007/978-1-59745-399-8_19

30. Shen Y, Xiao K, Liang P, Sun J, Sai S, Huang X. Characterization of soluble microbial products in 10 large-scale membrane bioreactors for municipal wastewater treatment in China. *J Memb Sci*. 2012;415-416:336-345. doi:10.1016/J.MEMSCI.2012.05.017

31. Melin T, Jefferson B, Bixio D, et al. Membrane bioreactor technology for wastewater treatment and reuse. *Desalination*. 2006;187(1-3):271-282. doi:10.1016/J.DESAL.2005.04.086

32. Kalboussi N, Rapaport A, Bayen T, Ben Amar N, Ellouze F, Harmand J. Optimal control of a membrane filtration system. *IFAC-PapersOnLine*. 2017;50(1):8704-8709. doi:10.1016/J.IFACOL.2017.08.1554

33. Hernández-Castro JA, Li K, Meunier A, Juncker D, Veres T. Fabrication of large-area polymer microfilter membranes and their application for particle and cell enrichment. *Lab Chip*. 2017;17(11):1960-1969. doi:10.1039/C6LC01525E

34. Luo Y, Zare RN. Perforated membrane method for fabricating three-dimensional polydimethylsiloxane microfluidic devices. *Lab Chip*. 2008;8(10):1688-1694. doi:10.1039/b807751g

35. Wei H, Chueh B-H, Wu H, et al. Particle sorting using a porous membrane in a microfluidic device. *Lab Chip*. 2011;11(2):238-245. doi:10.1039/c0lc00121j

36. Nagrath S, Sequist L V, Maheswaran S, et al. Isolation of rare circulating tumour cells in cancer patients by microchip technology. *Nature*. 2007;450(7173):1235-1239. doi:10.1038/nature06385

37. Di Carlo D, Edd JF, Irimia D, Tompkins RG, Toner M. Equilibrium separation and filtration of particles using differential inertial focusing. *Anal Chem*. 2008;80(6):2204-2211. doi:10.1021/ac702283m

38. Didar TF, Tabrizian M. Adhesion based detection, sorting and enrichment of cells in microfluidic Lab-on-Chip devices. *Lab Chip*. 2010;10(22):3043. doi:10.1039/c0lc00130a

39. Lapierre V, Pellegrini N, Bardey I, et al. Cord blood volume reduction using an automated system (Sepax) vs. a semi-automated system (Optipress II) and a manual method (hydroxyethyl starch sedimentation) for routine cord blood banking: a comparative study. *Cytotherapy*. 2007;9(2):165-169. doi:10.1080/14653240701196811

40. Carstensen F, Kasperidus T, Wessling M. Overcoming the drawbacks of microsieves with micromeshes for in situ product recovery. *J Memb Sci*. 2013;436:16-27. doi:10.1016/j.memsci.2013.01.017

41. Heinzman JM, Rice SD, Corkan LA. Robotic Liquid Handlers and Semiautomated Cell Quantification Systems Increase Consistency and Reproducibility in High-Throughput, Cell-Based Assay. *J Lab Autom*. 2010:7-14. https://www.navitar.com/pdf/PTI_LAC_JALAPublication.pdf. Accessed February 16, 2015.

42. Kang Y-T, Doh I, Cho Y-H. Tapered-slit membrane filters for high-throughput viable circulating tumor cell isolation. *Biomed Microdevices*. 2015;17(2):45. doi:10.1007/s10544-015-9949-6

43. Zhou X, Zhao Y, Chen S, et al. Self-Assembly of pH-Responsive Microspheres for Intestinal Delivery of Diverse Lipophilic Therapeutics. *Biomacromolecules*. 2016;17(8):2540-2554. doi:10.1021/acs.biomac.6b00512

44. Secret E, Kelly SJ, Crannell KE, Andrew JS. Enzyme-Responsive Hydrogel Microparticles for Pulmonary Drug Delivery. *ACS Appl Mater Interfaces*. 2014;6(13):10313-10321. doi:10.1021/am501754s

45. Borisov SM, Wolfbeis OS. Temperature-Sensitive Europium(III) Probes and Their Use for Simultaneous Luminescent Sensing of Temperature and Oxygen. *Anal Chem*. 2006;78(14):5094-5101. doi:10.1021/ac060311d

46. Ding Z, Liu P, Hu D, et al. Redox-responsive dextran based theranostic nanoparticles for near-infrared/magnetic resonance imaging and magnetically targeted photodynamic therapy. *Biomater Sci*. 2017;5(4):762-771. doi:10.1039/C6BM00846A

47. McAteer MA, Sibson NR, von Zur Muhlen C, et al. In vivo magnetic resonance imaging of acute brain inflammation using microparticles of iron oxide. *Nat Med*. 2007;13(10):1253-1258. doi:10.1038/nm1631

48. Alexandra S. Angelatos, Benno Radt and, Caruso* F. Light-Responsive Polyelectrolyte/Gold Nanoparticle Microcapsules. 2005. doi:10.1021/JP045070X

49. Huynh E, Lovell JF, Helfield BL, et al. Porphyrin Shell Microbubbles with Intrinsic Ultrasound and Photoacoustic Properties. *J Am Chem Soc*. 2012;134(40):16464-16467. doi:10.1021/ja305988f

50. Woo Sun Shim †, Jae Sun Yoo †, You Han Bae ‡ and, Doo Sung Lee* †. Novel Injectable pH and Temperature Sensitive Block Copolymer Hydrogel. 2005. doi:10.1021/BM050521K

51. Pacheco PM, LE B, White D, Sulchek T. TUNABLE COMPLEMENT ACTIVATION BY PARTICLES WITH VARIABLE SIZE AND Fc DENSITY. *Nano Life*. 2013;3(2):1341001. doi:10.1142/S1793984413410018

52. Pacheco P, White D, Sulchek T. Effects of microparticle size and Fc density on macrophage phagocytosis. *PLoS One*. 2013;8(4):e60989. doi:10.1371/journal.pone.0060989

53. Holt BA, Bellavia MC, Potter D, White D, Stowell SR, Sulchek T. Fc microparticles can modulate the physical extent and magnitude of complement activity. *Biomater Sci*. 2017;5(3):463-474. doi:10.1039/C6BM00608F

54. Jhamb K. *Chromatography-Pharmaceuticals, Biotechnology | BCC Research*.; 2018. https://www.bccresearch.com/market-research/biotechnology/chromatography-in-pharmaceuticals-biotechnology-markets-report-bio163a.html. Accessed March 14, 2019.

55. Hostettmann K, Marston A, Hostettmann M. Separation Strategy and Combination of Methods. In: *Preparative Chromatography Techniques*. Berlin, Heidelberg: Springer Berlin Heidelberg; 1998:230-240. doi:10.1007/978-3-662-03631-0_10

56. Weber V, Ettenauer M, Linsberger I, et al. Functionalization and Application of Cellulose Microparticles as Adsorbents in Extracorporeal Blood Purification. *Macromol Symp*. 2010;294(2):90-95. doi:10.1002/masy.200900042

57. Seeger M, Sepúlveda C, Vallejos P, Aguillón JC, Ferreira A. A simple method for negative and positive selection of murine and human IgM-bearing lymphocytes based on the use of antibody-coated silica microparticles. *J Immunol Methods*. 1993;162(1):109-114. doi:10.1016/0022-1759(93)90412-Z

58. Fernández-Shaw S, Shorter SC, Naish CE, Barlow DH, Starkey PM. Isolation and purification of human endometrial stromal and glandular cells using immunomagnetic microspheres. *Hum Reprod*. 1992;7(2):156-161. doi:10.1093/oxfordjournals.humrep.a137609

59. Kim S, Sung D, Chang JH. Highly efficient antibody purification with controlled orientation of protein A on magnetic nanoparticles. *Medchemcomm*. 2018;9(1):108-112. doi:10.1039/C7MD00468K

60. Hostettmann K, Marston A, Hostettmann M. Separation of Macromolecules. In: *Preparative Chromatography Techniques*. Berlin, Heidelberg: Springer Berlin Heidelberg; 1998:202-216. doi:10.1007/978-3-662-03631-0_8

61. Lyczak JB, Cannon CL, Pier GB. Lung infections associated with cystic fibrosis. *Clin Microbiol Rev*. 2002;15(2):194-222. http://www.ncbi.nlm.nih.gov/pubmed/11932230. Accessed September 22, 2016.

62. Toone EJ. Bacterial infection remains a leading cause of death in both Western and developing world. Preface. *Adv Enzymol Relat Areas Mol Biol*. 2011;77:xi-xiii. http://www.ncbi.nlm.nih.gov/pubmed/21692365. Accessed January 9, 2018.

63. Ding L, Laurent JM, Jaffrin MY. Dynamic filtration of blood: a new concept for enhancing plasma filtration. *Int J Artif Organs*. 1991;14(6):365-370. http://www.ncbi.nlm.nih.gov/pubmed/1885245. Accessed August 8, 2016.

64. Salgado AJ, Coutinho OP, Reis RL. Bone Tissue Engineering: State of the Art and Future Trends. *Macromol Biosci*. 2004;4(8):743-765. doi:10.1002/mabi.200400026

65. Cross LM, Thakur A, Jalili NA, Detamore M, Gaharwar AK. Nanoengineered biomaterials for repair and regeneration of orthopedic tissue interfaces. *Acta Biomater*. 2016;42:2-17. doi:10.1016/J.ACTBIO.2016.06.023

66. Vavken P, Joshi S, Murray MM. TRITON-X is most effective among three decellularization agents for ACL tissue engineering. *J Orthop Res*. 2009;27(12):1612-1618. doi:10.1002/jor.20932

67. Mitra D, Whitehead J, Yasui OW, Leach JK. Bioreactor culture duration of engineered constructs influences bone formation by mesenchymal stem cells. *Biomaterials*. 2017;146:29-39. doi:10.1016/j.biomaterials.2017.08.044

68. Abousleiman RI, Sikavitsas VI. Bioreactors for Tissues of the Musculoskeletal System. In: *Tissue Engineering*. Boston, MA: Springer US; 2006:243-259. doi:10.1007/978-0-387-34133-0_17

69. Freed LE, Guilak F, Guo XE, et al. Advanced Tools for Tissue Engineering: Scaffolds, Bioreactors, and Signaling. *Tissue Eng*. 2006;12(12):3285-3305. doi:10.1089/ten.2006.12.3285

70. Vunjak-Novakovic G, Martin I, Obradovic B, et al. Bioreactor cultivation conditions modulate the composition and mechanical properties of tissue-engineered cartilage. *J Orthop Res*. 1999;17(1):130-138. doi:10.1002/jor.1100170119

71. Sladkova M, de Peppo G. Bioreactor Systems for Human Bone Tissue Engineering. *Processes*. 2014;2(2):494-525. doi:10.3390/pr2020494

72. Alvarez-Barreto JF, Linehan SM, Shambaugh RL, Sikavitsas VI. Flow Perfusion Improves Seeding of Tissue Engineering Scaffolds with Different Architectures. *Ann Biomed Eng*. 2007;35(3):429-442. doi:10.1007/s10439-006-9244-z

73. Jaffrin MY. Hydrodynamic Techniques to Enhance Membrane Filtration. *Annu Rev Fluid Mech*. 2012;44(July):77-96. doi:10.1146/annurev-fluid-120710-101112

74. Alvarez-Barreto JF, Linehan SM, Shambaugh RL, Sikavitsas VI. Flow Perfusion Improves Seeding of Tissue Engineering Scaffolds with Different Architectures. *Ann Biomed Eng*. 2007;35(3):429-442. doi:10.1007/s10439-006-9244-z

75. Kuntaegowdanahalli SS, Asgar A, Bhagat S, et al. Inertial microfluidics for continuous particle separation in spiral microchannels. *Lab Chip*. 2009;9(20):2973-

2980. doi:10.1039/b908271a

76. Wilding P, Kricka LJ, Cheng J, Hvichia G, Shoffner MA, Fortina P. Integrated cell isolation and polymerase chain reaction analysis using silicon microfilter chambers. *Anal Biochem*. 1998;257(2):95-100. doi:10.1006/abio.1997.2530

77. Ripperger S, Gösele W, Alt C, et al. Ullmann's Encyclopedia of Industrial Chemistry: Filtration, 1. Fundamentals. Ullmann's Encyclopedia of Industrial Chemistry. doi:10.1002/14356007.b02_10.pub3

78. Seader JD, Seider WD, Lewin DR, Boulle L, Rycrof A. *Separation Process Principles, 3rd Edition*.; 2006.

79. Wang Z, Ma J, Tang CY, Kimura K, Wang Q, Han X. Membrane cleaning in membrane bioreactors: A review. *J Memb Sci*. 2014;468:276-307. doi:10.1016/J.MEMSCI.2014.05.060

80. Holdich R. *Fundamentals of Particle Technology*. 1st ed. Midland Information Technology and Publishing; 2002. http://www.particles.org.uk/particle_technology_book/chapter_4.pdf. Accessed August 1, 2017.

81. Cogan NG, Li J, Badireddy AR, Chellam S. Optimal backwashing in dead-end bacterial microfiltration with irreversible attachment mediated by extracellular polymeric substances production. *J Memb Sci*. 2016;520:337-344. doi:10.1016/J.MEMSCI.2016.08.001

82. Akhondi E, Wicaksana F, Fane AG. Evaluation of fouling deposition, fouling reversibility and energy consumption of submerged hollow fiber membrane systems with periodic backwash. *J Memb Sci*. 2014;452:319-331. doi:10.1016/J.MEMSCI.2013.10.031

83. Smith PJ, Vigneswaran S, Ngo HH, Ben-Aim R, Nguyen H. Design of a generic control system for optimising back flush durations in a submerged membrane hybrid reactor. *J Memb Sci*. 2005;255:99-106. doi:10.1016/j.memsci.2005.01.026

84. Todd G. *Laboratory Automation: Technologies and Global Markets - IAS076A*. Wellesley, MA; 2013. http://www.bccresearch.com/market-research/instrumentation-and-sensors/laboratory-automation-market-ias076a.html. Accessed April 16, 2016.

85. Dewan SS. *Global Markets and Technologies for Cell and Tissue Analysis -- Focus on Cell and Tissue Separation: BIO117A | BCC Research*.; 2013.

https://www.bccresearch.com/market-research/biotechnology/cell-tissue-separation-bio117a.html.

86. Wyatt Shields IV C, Reyes CD, López GP. Microfluidic cell sorting: a review of the advances in the separation of cells from debulking to rare cell isolation. *Lab Chip*. 2015;15(5). doi:10.1039/C4LC01246A

87. Kalboussi N, Harmand J, Rapaport A, Bayen T, Ellouze F, Ben Amar N. Optimal control of physical backwash strategy - towards the enhancement of membrane filtration process performance. *J Memb Sci*. 2018;545:38-48. doi:10.1016/J.MEMSCI.2017.09.053

88. Cogan NG, Chellam S. A method for determining the optimal back-washing frequency and duration for dead-end microfiltration. *J Memb Sci*. 2014;469:410-417. doi:10.1016/J.MEMSCI.2014.06.052

89. Villarroel R, Delgado S, González E, Morales M. Physical cleaning initiation controlled by transmembrane pressure set-point in a submerged membrane bioreactor. *Sep Purif Technol*. 2013;104:55-63. doi:10.1016/J.SEPPUR.2012.10.047

90. Smith PJ, Vigneswaran S, Ngo HH, Ben-Aim R, Nguyen H. A new approach to backwash initiation in membrane systems. *J Memb Sci*. 2006;278:381-389. doi:10.1016/j.memsci.2005.11.024

91. Cogan NG, Chellam S. Global parametric sensitivity analysis of a model for dead-end microfiltration of bacterial suspensions. *J Memb Sci*. 2017;537:119-127. doi:10.1016/J.MEMSCI.2017.05.042

92. Meunier A, Hernadez-Castro JA, Turner K, Li K, Veres T, Juncker D. Combination of Mechanical and Molecular Filtration for Enhanced Enrichment of Circulating Tumor Cells. doi:10.1021/acs.analchem.6b01324

93. Mohamed H, Turner JN, Caggana M. Biochip for separating fetal cells from maternal circulation. *J Chromatogr A*. 2007;1162(2):187-192. doi:10.1016/J.CHROMA.2007.06.025

94. Tajbakhsh S, Hajiali F. A comprehensive study on the fabrication and properties of biocomposites of poly(lactic acid)/ceramics for bone tissue engineering. *Mater Sci Eng C*. 2017;70:897-912. doi:10.1016/J.MSEC.2016.09.008

95. Miller DJ, Kasemset S, Paul DR, Freeman BD. Comparison of membrane fouling at constant flux and constant transmembrane pressure conditions. 2014.

doi:10.1016/j.memsci.2013.12.027

96. Stoller M, Ochando-Pulido JM. About merging threshold and critical flux concepts into a single one: the boundary flux. *ScientificWorldJournal*. 2014;2014:656101. doi:10.1155/2014/656101

97. Sabia G, Ferraris M, Spagni A. Online monitoring of MBR fouling by transmembrane pressure and permeability over a long-term experiment. *Sep Purif Technol*. 2014;122:297-305. doi:10.1016/j.seppur.2013.11.022

98. Mugnier N, Howell JA, Ruf M. Optimisation of a back-flush sequence for zeolite microfiltration. *J Memb Sci*. 2000;175:149-161. https://mail-attachment.googleusercontent.com/attachment/u/0/?ui=2&ik=955c5f4c25&view=att&th=15d51a3e3429ca90&attid=0.1&disp=inline&safe=1&zw&saddbat=ANGjdJ8BDCBRY4Assw11FeIqG1KPOI24UsoMqzkXvOaEdBKmEvvODo9n5qobSr5u63S94BfeKh9nbloBzJUlTAEwK-QsqJhbjJhkmuE2-. Accessed August 3, 2017.

99. Rodgers VGJ, Sparks RE. Effect of transmembrane pressure pulsing on concentration polarization. *J Membr Sci Elsevier Sci Publ BV*. 1992;68(149):149-168. http://ac.els-cdn.com/037673889280158G/1-s2.0-037673889280158G-main.pdf?_tid=1045e514-77b4-11e7-8075-00000aab0f01&acdnat=1501700271_68e5007381ab0e449f5aac4c40ffbede. Accessed August 2, 2017.

100. Zeng Y-S, Fan H, Xu B, et al. A facile strategy to integrate robust porous aluminum foil into microfluidic chip for sorting particles. *Microfluid Nanofluidics*. 2017;21(12):173. doi:10.1007/s10404-017-2001-9

101. Aran K, Sasso LA, Kamdar N, Zahn JD. Irreversible, direct bonding of nanoporous polymer membranes to PDMS or glass microdevices. *Lab Chip*. 2010;10(5):548-552. doi:10.1039/b924816a

102. Sackmann EK, Fulton AL, Beebe DJ. The present and future role of microfluidics in biomedical research. *Nature*. 2014;507(7491):181-189. doi:10.1038/nature13118

103. Ho C, Zydney AL. A Combined Pore Blockage and Cake Filtration Model for Protein Fouling during Microfiltration. 2000;399:389-399. doi:10.1006/jcis.2000.7231

104. Enten A, Leipner M, Bellavia M, King L, Sulchek T. Optimizing Flux Capacity of Dead-end Filtration Membranes by Controlling Flow with Pulse Width Modulated Periodic Backflush. *Nat - Sci Reports*. 2019.

105. Pacheco P, White D, Sulchek T. Effects of Microparticle Size and Fc Density on Macrophage Phagocytosis. Wang Y, ed. *PLoS One*. 2013;8(4):e60989. doi:10.1371/journal.pone.0060989

106. Pacheco PM, Le B, White D, et al. TUNABLE COMPLEMENT ACTIVATION BY PARTICLES WITH VARIABLE SIZE AND Fc DENSITY. *Nano Life*. 2013;3(2):1341001. doi:10.1142/S1793984413410018

107. Tang JL, Schoenwald K, Potter D, White D, Sulchek T. Bifunctional Janus microparticles with spatially segregated proteins. *Langmuir*. 2012;28(26):10033-10039. doi:10.1021/la3010079

108. Holt BA, Bellavia MC, Potter D, White D, Stowell SR, Sulchek T. Fc microparticles can modulate the physical extent and magnitude of complement activity. *Biomater Sci*. 2017;5(3):463-474. doi:10.1039/C6BM00608F

109. Tang JL, Schoenwald K, Potter D, White D, Sulchek T. Bifunctional Janus microparticles with spatially segregated proteins. *Langmuir*. 2012;28(26):10033-10039. doi:10.1021/la3010079

110. Soper SA, Brown K, Ellington A, et al. Point-of-care biosensor systems for cancer diagnostics/prognostics. *Biosens Bioelectron*. 2006;21(10):1932-1942. doi:10.1016/j.bios.2006.01.006

111. Burns JL, Gibson RL, McNamara S, et al. Longitudinal Assessment of *Pseudomonas aeruginosa* in Young Children with Cystic Fibrosis. *J Infect Dis*. 2001;183(3):444-452. doi:10.1086/318075

112. Ramsey BW. Management of Pulmonary Disease in Patients with Cystic Fibrosis. Wood AJJ, ed. *N Engl J Med*. 1996;335(3):179-188. doi:10.1056/NEJM199607183350307

113. Armstrong DS, Grimwood K, Carlin JB, Carzino R, Olinsky A, Phenlan PD. Bronchoalveolar lavage or oropharyngeal cultures to identify lower respiratory pathogens in infants with cystic fibrosis. *Pediatr Pulmonol*. 1996;21(5):267-275. doi:10.1002/(SICI)1099-0496(199605)21:5<267::AID-PPUL1>3.0.CO;2-K

114. Rosenfeld M, Emerson J, Accurso F, et al. Diagnostic accuracy of oropharyngeal cultures in infants and young children with cystic fibrosis. *Pediatr Pulmonol*. 1999;28(5):321-328. http://www.ncbi.nlm.nih.gov/pubmed/10536062. Accessed January 18, 2018.

115. Ramsey BW, Wentz KR, Smith AL, et al. Predictive Value of Oropharyngeal

Cultures for Identifying Lower Airway Bacteria in Cystic Fibrosis Patients. *Am Rev Respir Dis*. 1991;144(2):331-337. doi:10.1164/ajrccm/144.2.331

116. Ziegler-Graham K, MacKenzie EJ, Ephraim PL, Travison TG, Brookmeyer R. Estimating the prevalence of limb loss in the United States: 2005 to 2050. *Arch Phys Med Rehabil*. 2008;89(3):422-429. doi:10.1016/j.apmr.2007.11.005

117. Hooks JST, Clement CC, Nguyen H-D, Santambrogio L, Dixon JB. In vitro model reveals a role for mechanical stretch in the remodeling response of lymphatic muscle cells. *Microcirculation*. 2019;26(1):e12512. doi:10.1111/micc.12512

118. Charcosset C. Membrane processes in biotechnology: An overview. *Biotechnol Adv*. 2006;24(5):482-492. doi:10.1016/J.BIOTECHADV.2006.03.002

119. Gülle S, Bak M, Serdaroglu E, Can D, Karabay O. Low-Density Lipoprotein Apheresis by Membrane Differential Filtration (Cascade Filtration) via Arteriovenous Fistula Performed in Children With Familial Hypercholesterolemia. *Ther Apher Dial*. 2010;14(1):87-92. doi:10.1111/j.1744-9987.2009.00795.x

120. Jalilvand Z, Zokaee Ashtiani F, Fouladitajar A, Rezaei H. Computational fluid dynamics modeling and experimental study of continuous and pulsatile flow in flat sheet microfiltration membranes. *J Memb Sci*. 2014;450:207-214. doi:10.1016/j.memsci.2013.09.008